Lecture Notes in Computer Science 5545

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Gabrielle Allen Jarosław Nabrzyski
Edward Seidel Geert Dick van Albada
Jack Dongarra Peter M.A. Sloot (Eds.)

Computational Science – ICCS 2009

9th International Conference
Baton Rouge, LA, USA, May 25-27, 2009
Proceedings, Part II

Springer

Volume Editors

Gabrielle Allen
Jarosław Nabrzyski
Louisiana State University
Center for Computation & Technology
216 Johnston Hall, Baton Rouge, LA 70803, USA
E-mail: {gallen, naber}@cct.lsu.edu

Edward Seidel
Louisiana State University
Department of Physics and Astronomy
202 Nicholson Hall, Baton Rouge, LA 70803, USA
E-mail: eseidel@lsu.edu

Geert Dick van Albada
Peter M.A. Sloot
University of Amsterdam
Faculty of Science
Section Computational Science
Science Park 107, 1098 XG Amsterdam, The Netherlands
E-mail: {G.D.vanAlbada, P.M.A.Sloot}@uva.nl

Jack Dongarra
University of Tennessee
Computer Science Department
Knoxville, TN 37996-3450, USA
E-mail: dongarra@eecs.utk.edu

Library of Congress Control Number: Applied for

CR Subject Classification (1998): F, D.2, D.3, G.1, I.6, I.4, I.3, K.3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-642-01972-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-01972-2 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12687749 06/3180 5 4 3 2 1 0

Preface

"There is something fascinating about science.
One gets such wholesale returns of conjecture
out of such a trifling investment of fact."

Mark Twain, *Life on the Mississippi*

The challenges in succeeding with computational science are numerous and deeply affect all disciplines. NSF's 2006 Blue Ribbon Panel of Simulation-Based Engineering Science (SBES)[1] states 'researchers and educators [agree]: computational and simulation engineering sciences are fundamental to the security and welfare of the United States...We must overcome difficulties inherent in multiscale modeling, the development of next-generation algorithms, and the design...of dynamic data-driven application systems...We must determine better ways to integrate data-intensive computing, visualization, and simulation. Importantly, we must overhaul our educational system to foster the interdisciplinary study...The payoffs for meeting these challenges are profound.' The International Conference on Computational Science 2009 (ICCS 2009) explored how computational sciences are not only advancing the traditional hard science disciplines, but also stretching beyond, with applications in the arts, humanities, media and all aspects of research. This interdisciplinary conference drew academic and industry leaders from a variety of fields, including physics, astronomy, mathematics, music, digital media, biology and engineering. The conference also hosted computer and computational scientists who are designing and building the cyber infrastructure necessary for next-generation computing. Discussions focused on innovative ways to collaborate and how computational science is changing the future of research. ICCS 2009: 'Compute. Discover. Innovate.' was hosted by the Center for Computation and Technology at Louisiana State University in Baton Rouge. Talks and presentations at this conference focused on new applications for high-performance computing, including petascale algorithms, tools and applications, high-speed optical networks such as the Louisiana Optical Network Initiative, or LONI, distributed data management and sharing, and new software programs for biomedical, science and humanities research. The conference included tutorials, a main track session with 5 keynote speakers and 60 accepted, peer-reviewed papers as well as 13 workshops with 138 accepted peer-reviewed papers. Advancing computational science would not be possible without engaging students and young scholars. Through participation in tutorials, workshop and general session paper presentations, the students learned about recent advances and developments in computational science. This year ICCS 2009

[1] Blue Ribbon Panel Report: `www.nsf.gov/pubs/reports/sbes_final_report.pdf`

co-funded with the National Science Foundation a conference student scholarship for around 50 students, mostly from the state of Louisiana. ICCS is committed to helping students and young researchers enhance their professional development through participation in ICCS. During this year's conference two different tutorials were offered to participants: (i) Parallel Performance Evaluation Tools for HPC Systems by Allen D. Malony, University of Oregon, Markus Geimer, FZ Jülich, Andreas Knüpfer, TU Dresden, and Rick Kufrin, NCSA/University of Illinois and (ii) Developing HPC Applications with the Cactus Framework by Erik Schnetter, Frank Loeffler, Eloisa Bentivegna, CCT-LSU. The general main track of ICSS 2009 was organized in about 20 parallel sessions addressing the following topics:

- e-Science Applications and Systems
- Scheduling
- Software Services and Tools
- New Hardware and Its Applications
- Computer Networks
- Simulation of Complex Systems
- Image Processing
- Optimization Techniques
- Numerical Methods

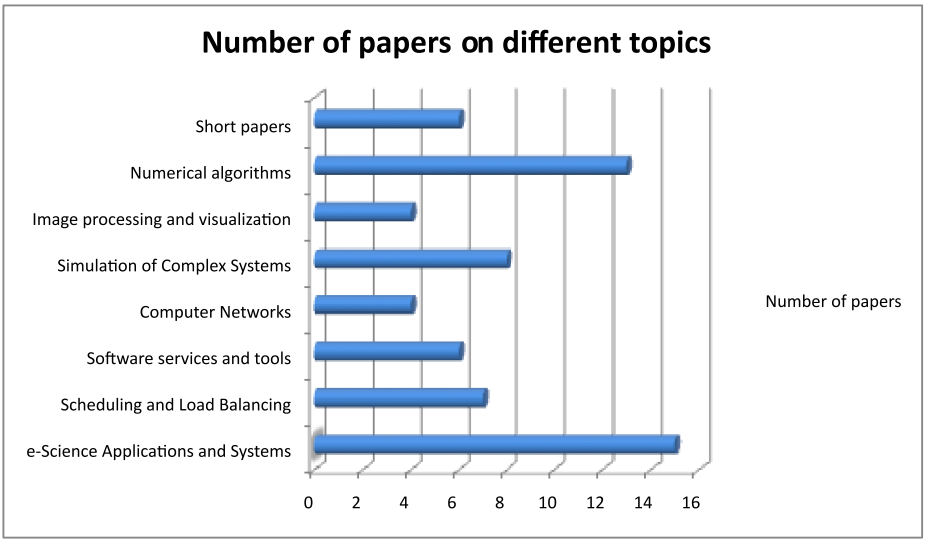


Fig. 1. Number of papers in the general track by topic

Figure 1 presents the number of papers on different topics.
Keynote lectures were delivered by:

- Marian Bubak: *Environments for collaborative applications: An answer to computational science challenges?*

– Janice Coen: *Computational modeling of wildland fire behavior and weather for research and forecasting,*
– Vittoria Colizza: *Computational epidemiology: a new paradigm in the fight against infectious diseases,*
– Peter Coveney: *Grid computing at the petascale,*
– Mark Jarrell: *Massively parallel and multi-scale simulations of strongly correlated electronic systems*

We would like to thank all keynote speakers for their interesting and inspiring talks and for submitting the abstracts and papers for this proceedings volume.

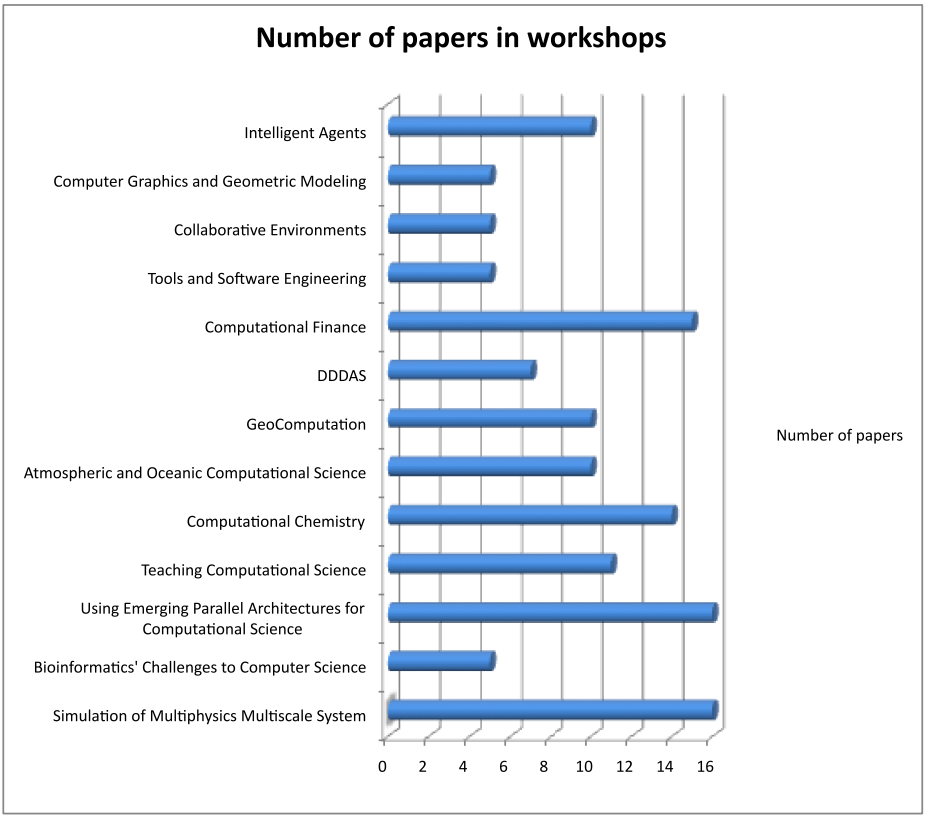


Fig. 2. Number of papers in workshops

The conference also offered 13 workshops:

– Teaching Computational Science
– Computational Chemistry and Its Applications
– Dynamic Data-Driven Application Systems
– Tools for Program Development and Analysis in Computational Science and Software Engineering for Large-Scale Computing

- Simulation of Multiphysics Multiscale Systems
- Workshop on Computational Finance and Business Intelligence
- Bioinformatics' Challenges to Computer Science
- Using Emerging Parallel Architectures for Computational Science
- Collaborative and Cooperative Environments
- Computer Graphics and Geometric Modeling
- Intelligent Agents in Simulation and Evolvable Systems
- Atmospheric and Oceanic Computational Science
- Geo Computation

Figure 2 presents the number of papers in the workshops.

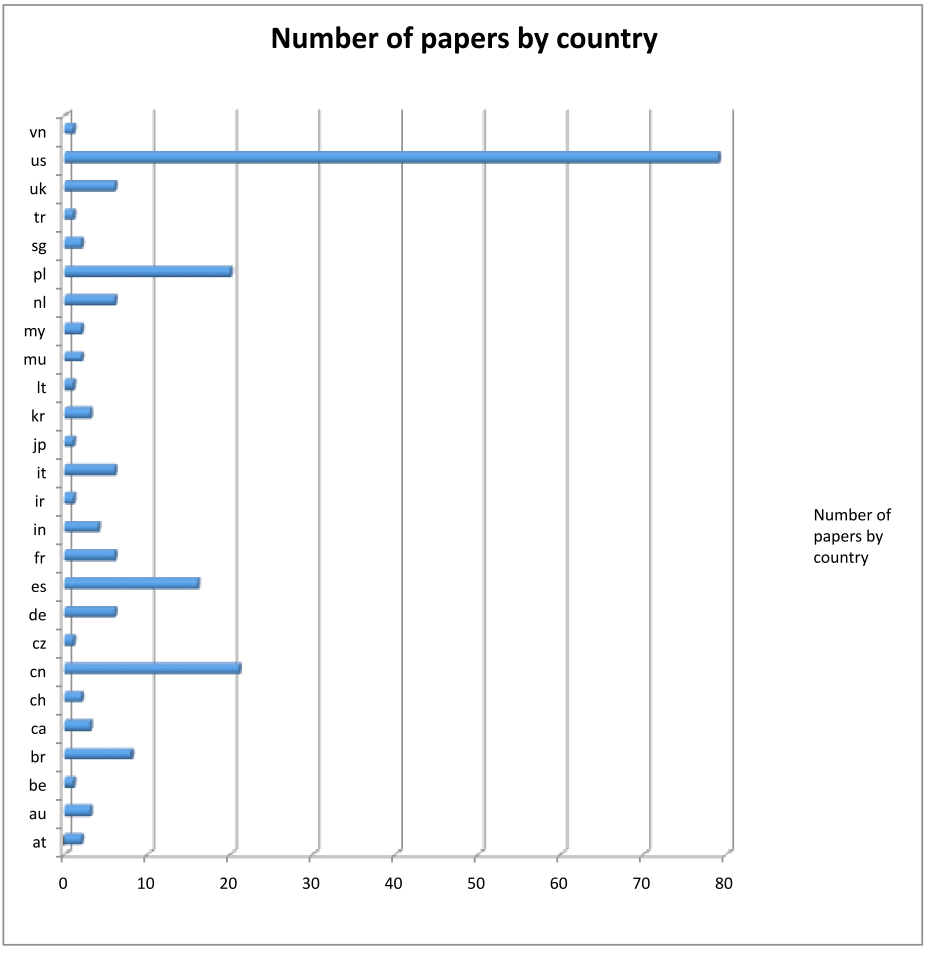


Fig. 3. Number of accepted papers by country

The selection of papers for the conference was possible thanks to the hard work of the Program Committee members and about 390 reviewers; papers submitted to ICCS 2009 received three reviews each. ICCS is a truly international conference, and papers were accepted from 26 countries. The international distribution of papers accepted for the conference is presented in Fig. 3. The ICCS 2009 participants equally represent all continents.

The ICCS 2009 proceedings consist of two volumes; the first one, LNCS 5544, contains the contributions presented in the general track and workshops 5, 7 and 12, while volume LNCS 5545 contains papers accepted for the other workshops. We hope that the ICCS 2009 proceedings will serve as an important intellectual resource for computational and computer science researchers, pushing forward the boundaries of these two fields and enabling better collaboration and exchange of ideas. We would like to thank Springer for a very fruitful collaboration during the preparation of the proceedings.

At the conference the best papers from the general track and workshops were nominated and presented on the ICCS 2009 website; the awards were funded by Elsevier. A number of papers will also be published as special issues of selected journals.

We owe thanks to all workshop organizers and members of the Program Committee for their diligent work, which has ensured the very high quality of the ICCS 2009. We are indebted to all the members of the Local Organizing Committee for their enthusiastic work towards the success of ICCS 2009, and to numerous colleagues from CCT for their help in editing the proceedings and organizing the event. We owe thanks to the ICCS 2009 sponsors: Intel, SiCortex, NSF, Elsevier, CCT and LSU Foundation for their generous support.

We wholeheartedly invite you to once again visit the ICCS 2009 website (http://www.iccs-meeting.org/iccs2009/), to recall the atmosphere of those May days in Louisiana.

May 2009

Gabrielle Allen
Jarek Nabrzyski
Ed Seidel
G. Dick van Albada
Jack J. Dongarra
Peter M.A. Sloot

Organization

ICCS 2009 was organized by the Center for Computation and Technology, Louisiana State University (Baton Rouge, USA), the University of Amsterdam (Amsterdam, The Netherlands) and the University of Tennessee (Knoxville, USA). All the members of the Local Organizing Committee are staff members of the Center for Computation and Technology, Louisiana State University.

Conference Chairs

Conference Chair – Ed Seidel (Center for Computation and Technology, Louisiana State University, USA)
Conference Co-chair – Gabrielle Allen (Center for Computation and Technology, Louisiana State University, USA)
Conference Co-chair – Jarek Nabrzyski (Center for Computation and Technology, Louisiana State University, USA)
Workshop Chair – Dick van Albada (University of Amsterdam, The Netherlands)
Overall Scientific Co-chair – Jack Dongarra (University of Tennessee, USA)
Overall Scientific Chair – Peter Sloot (University of Amsterdam, The Netherlands)

Local Organizing Committee

Gabrielle Allen
Ashlen Boudreaux
Jennifer Claudet
Karen Jones
Jarek Nabrzyski
Susie Poskonka
Kristen Sunde
Debra Waters
Adam Yates
Beata Nabrzyska
Lena Lacińska

Sponsoring Institutions

Intel Corporation
SiCortex
National Science Foundation
Elsevier
CCT
LSU Foundation

Program Committee

J.H. Abawajy	Deakin University, Australia
D. Abramson	Monash University, Australia
G.D. Allen	Louisiana State University, USA
I. Altintas	San Diego Supercomputer Centre, UCSD, USA
M. Antolovich	Charles Stuart University, Australia
E. Araujo	Universidade Federal de Campina Grande, Brazil
E. Bagheri	University of New Brunswick, Canada
B. Baliś	AGH University of Science and Technology, Kraków, Poland
P.K. Baruah	Sri Sathya Sai University, India
A. Benoit	LIP, ENS Lyon, France
I. Bethke	University of Amsterdam, The Netherlands
J. Bi	Tsinghua University, Beijing, China
J.A.R. Blais	University of Calgary, Canada
M. Bubak	AGH University of Science and Technology, Kraków, Poland
K. Bubendorfer	Victoria University of Wellington, New Zealand
J. Buisson	Institut TELECOM, Universite europeenne de Bretagne, France
J. Chen	Swinburne University of Technology, Australia
J. Cui	University of Amsterdam, The Netherlands
J.C. Cunha	Universidade Nova de Lisboa, Portugal
S. Date	Osaka University, Japan
S. Deb	National Institute of Science and Technology, Berhampur, India
B. Depardon	Universit de Lyon - ENS - LIP, France
T. Dhaene	Ghent University, Belgium
I.T. Dimov	University of Reading and IPP, Bulgarian Academy of Sciences, Bulgaria
J. Dongarra	University of Tennessee, USA
F. Donno	CERN, Switzerland
V. Duarte	Universidade Nova de Lisboa, Portugal
J. Ferreira da Silva	Universidade Nova de Lisboa, Portugal
G. Fox	Indiana University, USA
W. Funika	AGH University of Science and Technology, Kraków, Poland
B. Glut	AGH University of Science and Technology, Kraków, Poland
Y. Gorbachev	St. Petersburg State Polytechnical University, Russia

A.M. Gościński	Deakin University, Australia
G.A. Gravvanis	Democritus University of Thrace, Greece
D.J. Groen	University of Amsterdam, The Netherlands
T. Gubała	ACC CYFRONET AGH, Kraków, Poland
M. Hardt	Forschungszentrum Karlsruhe, Germany
T. Heinis	ETH Zurich, Switzerland
L. Hluchý	Institute of Informatics SAS, Slovakia
H. Jin	Huazhong University of Science and Technlogy, China
D. Johnson	ACET Centre, University of Reading, UK
B.D. Kandhai	University of Amsterdam, The Netherlands
S. Kawata	Utsunomiya University, Japan
W.A. Kelly	Queensland University of Technology, Australia
J. Kitowski	Inst.Comp.Sci. AGH-UST, Kraków, Poland
M. Koda	University of Tsukuba, Japan
D. Kranzlmüller	LMU and LRZ, Munich, Germany
B. Kryza	Academic Computer Centre CYFRONET-AGH, Kraków, Poland
L. Lefèvre	INRIA, France
A. Lewis	Griffith University, Australia
H.W. Lim	SAP Research, France
E. Lorenz	University of Amsterdam, The Netherlands
P. Lu	University of Alberta, Canada
M. Malawski	AGH University of Science and Technology, Kraków, Poland
M. Mascagni	Florida State University, USA
A.S. McGough	London e-Science Centre, UK
J. Nabrzyski	Louisiana State University, USA
S. Naqvi	CETIC, Belgium
P.O.A. Navaux	Universidade Federal do Rio Grande do Sul, Brazil
Z. Németh	MTA SZTAKI Computer and Automation Research Institute, Budapest, Hungary
B. Ó Nualláin	University of Amsterdam, The Netherlands
M. Paprzycki	IBS PAN and WSM, Poland
C. Pautasso	University of Lugano, Switzerland
V. Prasanna	University of Southern California, USA
M.R. Radecki	ACK CYFRONET AGH, Poland
A. Rendell	Australian National University, Australia
M. Riedel	Forschungszentrum Jülich, Germany
D. Rodríguez García	University of Alcalá, Spain
K. Rycerz	AGH University of Science and Technology, Kraków, Poland
B. Schulze	LNCC, Brazil
J. Seo	University of Leeds, UK

R. Slota	AGH University of Science and Technology, Kraków, Poland
A.E. Solomonides	University of the West of England, Bristol, UK
V.S. Stankovski	University of Ljubljana, Slovenia
A. Streit	Forschungszentrum Jülich, Germany
H. Sun	Beihang University, China
R. Tadeusiewicz	AGH University of Science and Technology, Kraków, Poland
C. Tedeschi	INRIA, France
A. Tirado-Ramos	University of Amsterdam, The Netherlands
A. Tsinakos	Department of Industrial Informatics, T.E.I. of Kavala, Greece
P. Tvrdik	Czech Technical University Prague, Chech Republic
G.D. van Albada	University of Amsterdam, The Netherlands
S.J. van Albada	University of Sydney, Australia
D.W. Walker	Cardiff University, UK
C.L. Wang	University of Hong Kong, China
A.L. Wendelborn	University of Adelaide, Australia
Y. Xue	Chinese Academy of Sciences, China
F.-P. Yang	Chongqing University of Posts and Telecommunications, China
L.T. Yang	St. Francis Xavier University, Canada
C.T. Yang	Tunghai University, Taiwan
E.V. Zudilova-Seinstra	University of Amsterdam, The Netherlands

Reviewers

J.H. Abawajy	P.K. Baruah	B. Boghosian
D. Abramson	S. Battiato	F. Bongiovanni
M. Aldinucci	M. Baumgartner	S. Boriah
M. Alexe	S. Bayan	A. Brabazon
G.D. Allen	P. Bekaert	R. Brito
I. Altintas	N. Bell	W. Bronsvoort
S. Ambroszkiewicz	A. Benoit	M. Bubak
F. Andre	N. Bergmann	K. Bubendorfer
C. Anthes	H. Berman	J. Buisson
M. Antolovich	J. Bernsdorf	A. Byrski
E. Araujo	I. Bethke	V. Camps
E. Bagheri	A. Beugnard	M. Cannataro
B. Balis	H. Beyer	U. Catalyurek
G. Barnett	J. Bi	K. Cetnarowicz
G. Barone	J.A.R. Blais	T. Chai
L.P.S. Barra	P. Blowers	M. Chakravarty
L.J. Bartolotti	A. Bode	J. Chen

Z. Chen
Z.X. Chen
X. Chi
B. Chopard
Z. Cinar
A. Clark
T. Clark
E. Constantinescu
E. Conwell
C. Cotta
A. Craik
J. Cui
J.C. Cunha
A.C. da Rocha Costa
D. Daescu
F. Darema
K.K. Das
S. Date
B.R. De
J.N. de Souza
S. Deb
Y. Demazeau
B. Depardon
R. Dew
T. Dhaene
I.T. Dimov
G. Dobrowolski
T. Dokken
J. Dongarra
F. Donno
C.C. Douglas
R. Drezewski
S. Dua
V. Duarte
W. Dubitzky
F. Dufoss
S. Emrich
V. Ervin
K. Evans
D. Fedorov
J. Ferreira da Silva
T. Ford
G. Fox
F. Freitag
H. Freitas
C. Froidevaux
W. Funika
S. Furin
L.M. Gadelha Junior
M. Gallet
A. Galvez
A. Garny
T. Gedeon
A. Gerbessiotis
M. Gerrits
A. Giesler
S. Gimelshein
D. Gimenez
S. Girtelschmid
C. Glasner
B. Glut
R. Goh
L. Gorb
Y. Gorbachev
A.M. Gościński
A. Goursot
G.A. Gravvanis
C. Grelck
D.J. Groen
P. Gruer
T. Gubała
C. Guerra
X. Guo
Y. Guo
P.H. Guzzi
A. Haffegee
F.B.H. Hagelberg.
M. Hamada
U. Hansmann
M. Hardt
W.W. Hargrove
J. He
T. Heinis
P. Heinzlreiter
M. Hell
D. Henze
V. Hernández
P. Herrero
L. Hluchý
B. Hnatkowska
A. Hoekstra
F.M. Hoffman
J. Huang
M. Huebner
E. Hunt
A. Iglesias
T. Iliescu
D.A. Ivanov
H. Iwasaki
R. Jamieson
M. Jardak
A. Jashki
H. Jin
D. Johnson
J.J. Johnstone
T. Jurczyk
J. Jurek
R. Kakkar
A. Kalyanaraman
B.D. Kandhai
S. Kawata
W.P. Kegelmeyer
R. Kelly
W.A. Kelly
C. Kessler
C.H. Kim
M. Kisiel-Dorohinicki
J. Kitowski
C.R. Kleijn
A. Knüpfer
R. Kobler
M. Koda
I. Kolingerova
J. Kolodziej
R. Kooima
M. Korek
S. Kostov
G. Kou
J. Koźlak
P. Kozyra
M. Krafczyk
D. Kranzlmüller
B. Kryza
V.V. Krzhizhanovskaya
V. Kumar

A. Lagana
K.K. Lai
R. Landertshamer
D. Lavenier
H. Lee
H.K. Lee
H.S. Lee
L. Lefevre
P. Leong
A. Lewis
A.H. Li
G.Q. Li
J.P. Li
S. Li
H.W. Lim
Z. Lin
L. Liquori
B. Liu
D.S. Liu
J. Liu
R. Liu
W. Liu
Y. Liu
M. Lobosco
R. Loogen
E. Lorenz
P. Ltstedt
M. Low
P. Lu
D. Luebke
W. Luo
C. Lursinsap
T.J. Ma
S. MacLachlan
K. Madduri
N. Maillard
M. Makki
M. Malawski
U. Maran
M. Mascagni
D.L. Maskell
R. Mason
K. Matsuzaki
F. Mavelli
M.L. McCarthy
A.S. McGough
W. Meira
A.S. Memon
Z. Michalewicz
J. Michopoulos
S. Midkiff
R.T. Mills
A. Misra
D.G. Mitmik
G. Moltó
F. Munoz
A.R. Mury
J. Nabrzyski
R.D. Nair
S. Naqvi
A. Nasri
P.O.A. Navaux
E. Nawarecki
Z. Németh
L. Neumann
G. Nikishkov
B. Ó Nualláin
J.T. Oden
A. Oliferenko
D. Olson
M. O'Neill
P. Orantek
G. Ostrouchov
J. Owen
M. Paprzycki
C. Pautasso
B. Payne
T. Peachey
Y. Peng
M.S. Pérez
G. Pfeiffer
D. Plemenos
M. Polak
V. Prasanna
G. Qiu
A. Queiruga Dios
M.R. Radecki
U.R. Radius
B. Raffin
P. Ramasami
A. Rendell
M. Riedel
R. Righi
S. Ringuette
Y. Robert
E.R. Rodrigues
D. Rodríguez García
C. Rodríguez-Leon
F. Rogier
F.-X. Roux
R. Ruiz
M. Rumi
K. Rycerz
A. Sandu
M. Sbert
R. Schaefer
H.F. Schaefer III
J. Schatz
M. Schimmler
B. Schmidt
L. Schnorr
H. Schroder
J. Schroeder
B. Schulze
S. See
M. Segarre
J. Seo
A. Sfarti
H. Shi
Y. Shi
A. Shiflet
F. Silvestri
B. Simo
D. Sinclair
V. Sipkova
P. Slizik
R. Slota
B. Śnieżyński
A. Soldera
A.E. Solomonides
A. Sourin
R. Spiteri
V. Srovnal
A. Stamatakis
V.S. Stankovski

A. St-Cyr
A. Streit
J. Sumitomo
H. Sun
J. Sundnes
H. Suzuki
C. Swanson
C.T. Symons
D. Szczerba
L. Szirmay-Kalos
R. Tadeusiewicz
R. Tagliaferri
W.K. Tai
E. Talbi
T. Tang
X.J. Tang
J. Tao
M. Taufer
M. Taylor
C. Tedeschi
D. Thalmann
L. Tininini
A. Tirado-Ramos
R.F. Tong
A. Tsinakos
P. Tvrdik
B. Ucar
F. Ucun
G.V. Valenzano
A. Valladares
G.D. van Albada
S.J. van Albada
M. van der Hoef
R.R. Vatsavai
J. Villa i Freixa
J. Volkert
G. Voss
H.S. Wahab
D.W. Walker
C.L. Wang
J.J. Wang
M. Wang
Z.X. Wang
R. Weber dos Santos
W. Weijer
R. Wismüller
P.H. Worley
S.Y. Wu
Y. Xue
N. Yan
C.T. Yang
F.-P. Yang
L.T. Yang
J. Yu
I. Zelinka
J. Zhang
J.J. Zhang
L.L. Zhang
B. Zheng
A. Zhmakin
H.A. Zhong
X.F. Zhou
X. Zhu
J. Zola
A. Zomaya
E.V. Zudilova-Seinstra

Workshops Organizers

Third Workshop on Teaching Computational Science (WTCS 2009)

A.B. Shiflet (Wofford College, Spartanburg, SC), A. Tirado Ramos (University of Amsterdam)

Workshop on Computational Chemistry and Its Applications (4th CCA)

P. Ramasami (University of Mauritius), F.H. Schaefer III (University of Georgia)

Dynamic Data-Driven Application Systems - DDDAS 2009

C.C. Douglas (University of Wyoming)

Joint Workshop for Tools for Program Development and Analysis in Computational Science and Software Engineering for Large-Scale Computing

A. Knüpfer (ZIH, TU Dresden, Germany), D. Rodríguez (The University of Alcalá), J. Tao (Forschungszentrum Karlsruhe, Germany)

Simulation of Multiphysics Multiscale Systems, 6th International Workshop

V.V. Krzhizhanovskaya (University of Amsterdam), A.G. Hoekstra (University of Amsterdam)

Workshop on Computational Finance and Business Intelligence

Y. Shi (University of Nebraska at Omaha and Chinese Academy of Sciences), X. Deng (Department of Computer Science, City University of Hong Kong), S. Wang (Academy of Mathematical and System Sciences, Chinese Academy of Sciences)

Bioinformatics' Challenges to Computer Science

M. Cannataro (Magna Græcia of Catanzaro University), J. Sundnes (Simula Research Laboratory, Norway), R. Weber dos Santos (Federal University of Juiz de Fora, Brazil)

Using Emerging Parallel Architectures for Computational Science

B. Schmidt (Nanyang Technological University), D.L. Maskell (Nanyang Technological University)

Collaborative and Cooperative Environments

C. Anthes (GUP, Joh. Kepler University Linz), V. Alexandrov (ACET, University of Reading, UK), D. Kranzlmüller (LMU Munich and LRZ Garching, Germany), J. Volkert (GUP, Joh. Kepler University Linz)

Eighth International Workshop on Computer Graphics and Geometric Modeling, CGGM 2009

A. Iglesias (University of Cantabria)

Intelligent Agents in Simulation and Evolvable Systems

R. Schaefer (AGH University of Science and Technology), K. Cetnarowicz (AGH University of Science and Technology), B. Zheng (South-Central University For Nationalities, Wuhan, China)

Atmospheric and Oceanic Computational Science

A. Sandu (Virginia Tech), A. St-Cyr (National Center for Atmospheric Research), K. Evans (Oak Ridge National Laboratory)

Geo Computation

Y. Xue (London Metropolitan University), F.M. Hoffman (Oak Ridge National Laboratory), D.S. Liu (Remote-Sensing Satellite Ground Station, Chinese Academy of Sciences)

Table of Contents – Part II

Third Workshop on Teaching Computational Science (WTCS 2009)

Workshop on Computational Chemistry and Its Applications (4th CCA)

Workshop on Atmospheric and Oceanic Computational Science

Workshop on Geocomputation 2009

Workshop on Dynamic Data Driven Applications Systems – DDDAS 2009

Workshop on Computational Finance and Business Intelligence

Joint Workshop on Tools for Program Development and Analysis in Computational Science and Software Engineering for Large-Scale Computing

Workshop on Collaborative and Cooperative Environments

Eighth International Workshop on Computer Graphics and Geometric Modeling, CGGM 2009

Workshop on Intelligent Agents in Simulation and Evolvable Systems

Table of Contents – Part I

e-Science Applications and Systems

Scheduling and Load Balancing

Software Services and Tools

Computer Networks

Simulation of Complex Systems

Image Processing and Visualization

Numerical Algorithms

Short Papers

Simulation of Multiphysics Multiscale Systems, 6th International Workshop

Workshop on Bioinformatics' Challenges to Computer Science

Workshop on Using Emerging Parallel Architectures for Computational Science

Third Workshop on Teaching Computational Science (WTCS 2009)

Alfredo Tirado-Ramos[1] and Angela Shiflet[2]

[1] Informatics Institute, University of Amsterdam
Amsterdam, The Netherlands
[2] Wofford College
Spartanburg, South Carolina, U.S.A.
alfredo@science.uva.nl, shifletab@wofford.edu

Abstract. The Third Workshop on Teaching Computational Science, within the International Conference on Computational Science, provides a platform for discussing innovations in teaching computational sciences at all levels and contexts of higher education. This editorial provides an introduction to the work presented during the sessions.

Keywords: computational science, teaching, parallel computing, e-Learning, collaborative environments.

1 Introduction

Today's technology-driven societies require students who have been trained in technology-based environments, such as computational science. Until recently, computational science education was too costly and impractical for most academic institutions of higher learning. Now, however, such institutions can integrate methods from computer science, mathematical modeling, simulation, and scientific visualization, among others, to create virtual laboratories for in-silico experimentation and learning. The interaction of computational methods allows teachers and students to pose more intriguing questions in lower cost experimental settings. Thus, higher education is currently witnessing the rapid adoption of computational tools and methods by science teachers around the world. In the past few years, many teachers have shared experiences on the use of high performance, as well as not-so-high performance, computing facilities in order to promote the benefits and importance of computational science instruction in science classrooms. The Third International Workshop on Teaching Computational Science (WTCS2009), held in Baton Rouge, Louisiana, U.S.A., in conjunction with the International Conference on Computational Science (ICCS 2009) offers a technical program consisting of presentations dealing with the state of the art in the field, following the successful 2008 WTCS in Krakow, Poland [1]. The workshop includes presentations that describe innovations in the context of formal courses involving, for example, introductory programming, service courses and specialist undergraduate or postgraduate topics.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 3–4, 2009.

2 The Workshop

During the workshop sessions, we had 12 oral and 2 poster presentations. Nguyen et al discussed interesting approach to statistical methods to infer knowledge during the learning process. Landau proposes that teaching within this research-like, problem-solving approach is a more motivating and efficient technique than teaching the various disciplines separately. Jacobs presents an interesting experience and provides access to web material, which could be useful to prepare other similar programs. Bloomfield describes the use of R, a language typically used in computational science and engineering, as a useful tool in computational science teaching. Muresano describes a new innovative masters degree program with the aim of introducing students to core concepts such as large scale simulation and high performance computing. Shiflet describes successful strategies for managing epidemics in a microbiology course, where science students are usually unaware of potential advantages and complexities. Del Vado discusses an intelligent tutoring system for teaching data structures, an important subject in computational science training. Gavilanes describes an interesting tool for use in introductory classes, which allows students to focus on schemas and algorithm design rather than language syntax. Borne presents a computational science program centered on data representation and visualization covering a broad range of physical and biological sciences. Stevenson proposes extending classical argumentation structures as the basis for computational science education. Liu provides a summary of simulation and modeling process that illustrate how teaching tools can be used and how increasingly complex models can be introduced. Wallin presents the structure and goals of the first two years of a computational mathematics program, along with some observations about the elements that we have found that have been challenging in its implementation. Finally, as poster presentations, Gallardo presents an approach to adaptation to European Credit Transfer System, and Parker presents a new mathematics elective for an undergrad computational science program.

3 Conclusions

The presentations in ICCS 2009's Third International Workshop on Teaching Computational Science (WTCS2009) illustrate the variety and depth of computational science education around the world. Moreover, the growing success of the workshop attests to the increasing interest in this important interdisciplinary area.

Acknowledgments. We would like to acknowledge P.M.A. Sloot and D. van Albada for their continuous support and commitment to the success of this workshop.

Reference

1. Tirado-Ramos, A., Luo, Q.: Second Workshop on Teaching Computational Science WTCS 2008. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part II. LNCS, vol. 5102, pp. 657–658. Springer, Heidelberg (2008)

Combination of Bayesian Network and Overlay Model in User Modeling

Loc Nguyen[1] and Phung Do[2]

[1] University of Natural Science / Faculty of Information Technology,
Ho Chi Minh City, VietNam
[2] University of Information Technology / Faculty of Information System,
Ho Chi Minh City, VietNam
ng_phloc@yahoo.com

Abstract. The core of adaptive system is user model containing personal information such as knowledge, learning styles, goals which is requisite for learning personalized process. There are many modeling approaches, for example: stereotype, overlay, plan recognition… but they do not bring out the solid method for reasoning from user model. This paper introduces the statistical method that combines Bayesian network and overlay modeling so that it is able to infer user's knowledge from evidence collected during user's learning process.

Keywords: Bayesian network, overlay model, user model.

1 Introduction

User modeling is the new trend of enhancing the adaptability of e-learning system. User models are classified into: stereotype model, overlay model, differential model, perturbation model, plan model.

- Stereotype [27] is a set of user's frequent characteristics. In general, stereotype represents a category or group of learners.
- In overlay modeling, learner model is the subset of domain model. The domain is decomposed into a set of elements and the overlay model is simply a set of masteries over those elements.
- Differential model is basically an overlay on expected knowledge, which in turn is an overlay on expert's domain knowledge.
- Perturbation model represents learners as the subset of expert's knowledge plus their mal-knowledge.

Modeling user must follow three below steps:

- Initialization is the process that gathers information and data about user and constructs user model from this information.
- Updating intends to keep user model up-to-date.
- Reasoning new information about user out from available data in user model.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 5–14, 2009.

Reasoning is complex but essential and interesting, especially, there is need to deal with uncertain or imprecise information in user modeling. For example, answering the question: "The student failed the exam, so most probably he doesn't master the knowledge" is involved in processing uncertain information. The approaches which solve this problem primarily base on theory of artificial intelligence (AI) or statistics. Both AI and statistics have particular advantages and drawbacks but statistical method is appropriate to evaluate learner's performance by collecting evidence. Bayesian network which is the marriage between Bayesian inference and graph theory has a solid mathematical fundamental. Additionally, overlay model can represent very clearly user's knowledge.

In this paper, we propose the combination of overlay model and Bayesian network so that it is able to take full advantages of strong points of both of them.

Section 2: survey of Bayesian inference and Bayesian network, the core of our method. Section 3: Applying Bayesian network to overlay model. Section 4: Evaluation of this method and conclusion.

2 Bayesian Network

2.1 Bayesian Rule

Bayesian inference, a form of statistical method, is responsible for collecting evidence to change the current belief in given hypothesis. The more the observed evidence, the higher degree of belief in hypothesis is. First, this belief is assigned an initial probability. Note, in classical statistical theory, the random variable's probability is objective (physical) through trials. But, in Bayesian method, the probability of hypothesis is "personal" because its initial value is set subjectively by expert. When evidence is gathered enough, the hypothesis is considered trustworthy.

Bayesian inference is based on Bayesian rule with some special aspects:

$$P(H \mid E) = \frac{P(E \mid H) * P(H)}{P(E)} \tag{1}$$

H is probability variable denoting a hypothesis existing before evidence
E is also probability variable notating an observed evidence
P(H) is *prior probability* of hypothesis. It is also hypothesis' initial value
P(H | E), conditional probability of H with given E, is called *posterior probability*. It tell us the changed belief in hypothesis when occurring evidence
P(E | H) is conditional probability of occurring evidence E when hypothesis is true. In fact, likelihood ratio is P(E | H) / P(E) but P(E) is constant value. So we can consider P(E | H) as *likelihood function* of H with fixed E.
P(E) is probability of occurring evidence E together all mutually exclusive cases of hypothesis. If H and E are discrete, $P(E) = \sum_{H} P(E \mid H) * P(H)$, otherwise $f(e) = \int f(e \mid h) f(h) dh$ with h and e being continuous, f denoting probability density function. Because of being sum of products of prior probability and likelihood function, P(E) is called marginal probability.

Note: H, E must be random variables according to statistical theory.

2.2 Bayesian Network

Bayesian network is combination of graph theory and Bayesian inference. It having a set of nodes and a set of directed arcs is the directed acyclic graph (DAG). Each node represents a random variable which can be a evidence or hypothesis in Bayesian inference. Each arc reveals the cause-effect relationship among two nodes. If there is the arc from node A to B, we call "A causes B" or "A is the parent of B", in other words, A depends conditionally on B. Otherwise there is no arc between A and B, it asserts the conditional independence. Note, in Bayesian network context, terms: node and variable are the same.

A node has a local conditional probability distribution (CPD). If variables are discrete, CPD is simplified as table (CPT). When one node is conditionally dependent on another, there is a corresponding probability (in CPT or CPD) measuring the influence of causal node on it. In case node has no parent, its CPT degenerate into prior probabilities.

For example, in figure 1, event "cloudy" is cause of event "rain" or "sprinkler", which in turn is cause of "grass is wet". So we have three causal relationships of: 1-cloudy to rain, 2- rain to wet grass, 3- sprinkler to wet grass. This model is expressed below by Bayesian network with four nodes and three arcs corresponding to four events and three relationships. Every node has two possible values True (1) and False (0) together its CPT.

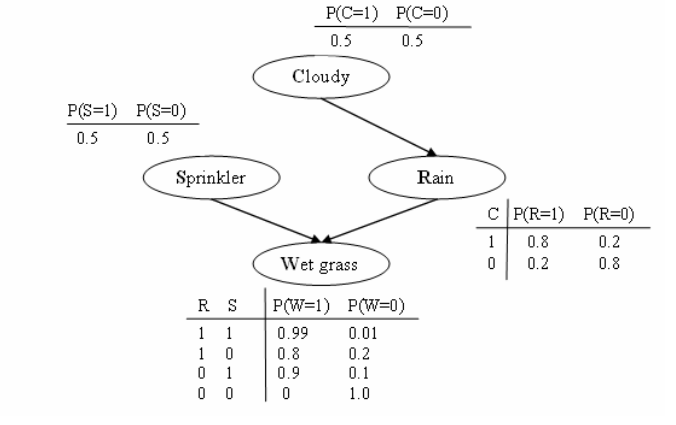


Fig. 1. Bayesian network (a classic example about "wet grass")

Suppose we use two letters x_i and $pa(x_i)$ to name a node and a set of its parent, correspondingly. X is vector which was constituted of all x_i, $X = (x_1, x_2,..., x_n)$. The global joint probability distribution $p(X)$ being product of all local CPDs or CPTs is formulated as:

$$p(x1, x2, ..., xn) = \prod_{i=1}^{n} p(xi \mid pa(xi)) \tag{2}$$

Suppose Ω_i is the subset of $pa(x_i)$ such that x_i must depend conditionally and directly on every variable in Ω_i. In other words, there is always an arc from each variable in Ω_i to x_i and no intermediate node between them.

Thus, formula 2 becomes:

$$p(x1, x2, ..., xn) = \prod_{i=1}^{n} p(xi \mid \Omega i) \tag{3}$$

In figure 1, according to formula 2:

$$p(C, R, S, W) = p(C)p(R \mid C)p(S \mid C)p(W \mid C, R, S)$$

Applying formula 3, $p(S \mid C) = p(S)$ due to the conditional independence assertion about variables S and C. Furthermore, because S is intermediate node between C and W, we should remove C from p(W | C, R, S), hence, $p(W \mid C, R, S) = p(W \mid R, S)$. In short, the expansion of formula 3 is shown below:

$$p(C,R,S,W) = p(C)p(S)p(R \mid C)p(W \mid R,S) \tag{4}$$

2.3 Inference in Bayesian Network

Using Bayesian reference, we need to compute the posterior probability of each hypothesis node in network. In general, the computation based on Bayesian rule is known as the inference in Bayesian network.

Reviewing figure 1, suppose W becomes evidence variable which is observed the fact that the grass is wet, so, W has value 1. There is request for answering the question: how to determine which cause (sprinkler or rain) is more possible for wet grass. Hence, we will calculate two posterior probabilities of S (=1) and R (=1) in condition W (=1). These probabilities are also called *explanations* for W.

$$p(R = 1 \mid W = 1) = \frac{\sum_{C,S} p(C, R = 1, S, W = 1)}{\sum_{C,R,S} p(C, R, S, W = 1)} \tag{5}$$

$$p(S = 1 \mid W = 1) = \frac{\sum_{C,R} p(C, R, S = 1, W = 1)}{\sum_{C,R,S} p(C, R, S, W = 1)} \tag{6}$$

In fact, formulas 5 and 6 are expansion of formula 1. Applying (4) to (5) & (6):

p(R=1|W=1)=0.4475/0.7695=0.581<p(S=1|W=1)=0.4725/0.7695=0.614

It is concluded that sprinkler is the most likely cause of wet grass.

3 Applying Bayesian Network to Overlay Model

The basic idea of overlay modeling is that the user model is the subset of domain model. Straightforward, the domain is decomposed into a set of knowledge elements and the overlay model (namely, user model) is simply a set of masteries over those elements. Suppose that the mastery of each element varies from 0 (*not mastered*) to

1 (*mastered*), to wit weighted overlay. The relationship of element A to element B is often prerequisite relationship, so, we can deduce that user must comprehend A before learning B. Then the expert model is the overlay with 1 for each element and the learner model is the overlay with at most 1 for each element.

Although overlay model is the simple but powerful method to represent user model, it does not provide the way to infer user's knowledge from evidence collected in user's learning process. Overlay modeling should associate with other statistical approach in solving this problem and Bayesian network is the best choice. So, we combined Bayesian network and overlay model by following steps:

1. The structure of overlay model is considered as Bayesian network. Thus, knowledge elements in domain become variables (or nodes) in Bayesian network. Instead of using the weight of each element as above, we assign the probability to each variable for estimating the mastery of knowledge. All variables are binary (0 – not mastered and 1 – mastered). Note, knowledge item, knowledge element and concept are synonymical terms.
2. The prerequisite relationships between knowledge elements are known as the conditional dependence assertions in Bayesian network. Accordingly, each node has a CPT.
3. All knowledge elements are defined as *hidden variables* (hypothesis). Other learning objects or events (such as: tests, exams, exercises, user's feedback, user's activities) which are used to assess or evaluate user's performance in learning process are consider as *evidence variables*. We must add them to Bayesian network along with determining the conditional dependence relationship between them and remaining hidden variable, namely, specifying their CPTs. Inferring user's knowledge is to compute posterior probability of hidden variables (according to formula 5, 6) when evidence variables change their values. This process can be known as knowledge diagnosis.

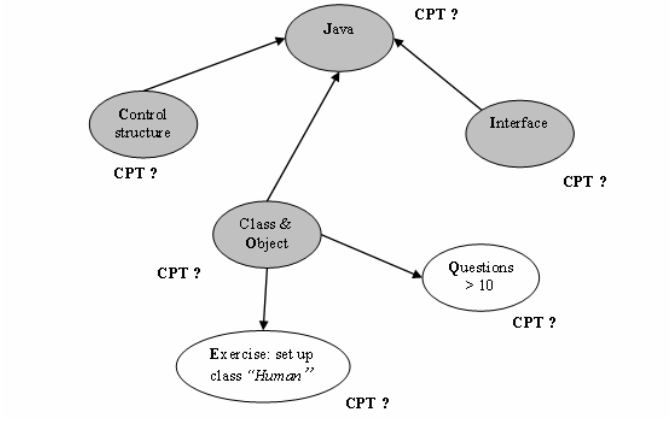


Fig. 2. Combination of Bayesian network and overlay model *(evidence nodes are unshaded, otherwise, hidden nodes are shaded)*

After completing the three steps described above, creating the learning network leaves us with two essential conditions:

- Specifying the structure of model including nodes and arcs. This task is development of qualitative model done by experts such as teachers, lecturers, supervisors… or by learning algorithms.
- Specifying the important parameters which are CPTs of all variables. This task called development of quantitative model is described in section 3.1.

3.1 Specifying CPTs of Variables

Suppose Java course is constituted of four concepts considered as hidden variables whose links are prerequisite relationships. Additionally, there are two evidence variables: "Questions > 10" and "Exercise: set up…". That learner asks more than 10 questions is to tell how much her/his amount of knowledge. Like that, evidence "Exercise: set up…" proves whether or not he/she understands concept "Class & Object". The number (in range 0…1) that measures the relative importance of each prerequisite or evidence is defined by expert or teacher. In other words, this is the *weight* of arc from parent node to child node. All weights concerning the child variable will build up its CPT. Sum of weights of all arcs to/from each child/parent node in case of hidden/evidence variable should be 1. It means that each weight is normalized.

Your attention please, the relationship between hidden variable (H) and evidence variable (E) must be from H to A because the process that computes posterior probability of hidden variable with evidence is the knowledge diagnosis. So, evidence variable has no child and its parents must be hidden variables. In short, there are two kinds of relationships:

- Prerequisite relationships among hidden variables.
- Diagnostic relationships of hidden variables to evidence. The mastery of concepts (hidden) effects on the trust of evidence. However, if learner failed an examination, it is not sure about her/his lack of knowledge or ability because she/he can make a mistake unexpectedly.

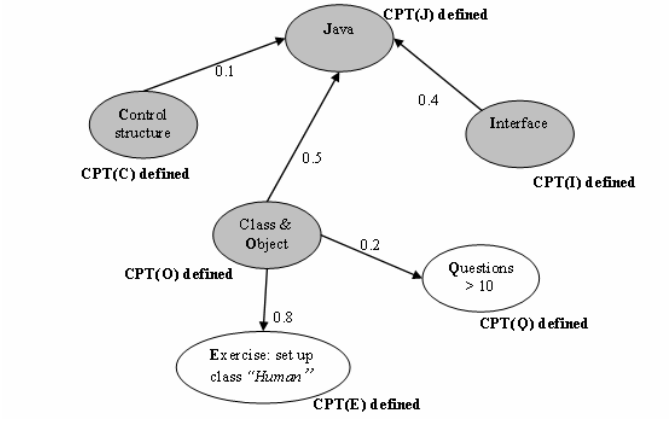


Fig. 3. Bayesian overlay model and its parameters in full *(evidence nodes are unshaded, otherwise, hidden nodes are shaded)*

In this example, node J (Java) has three parents: C (Control structure), O (Class & Object), I (Interface) which in turn are corresponding to three weights of prerequisite relationship: w_1=0.1, w_2=0.5, w_3=0.4. Conditional probability of J is computed as follows:

$$p(J \mid C, O, I) = w_1 * h_1 + w_2 * h_2 + w_3 * h_3$$

where $h1 = \begin{cases} 1 \ if \ C = J \\ 0 \ otherwise \end{cases}$ $h2 = \begin{cases} 1 \ if \ O = J \\ 0 \ otherwise \end{cases}$ $h3 = \begin{cases} 1 \ if \ I = J \\ 0 \ otherwise \end{cases}$

Note: {J, C, O, I} is complete set of mutually exclusive variables (of course, each also variable is random and binary). Generalizing about formula 7, it is that:

$$p(X = 1 \mid Y1, Y2, ..., Yn) = \sum_{i=1}^{n} wi * hi \tag{7}$$

where $hi = \begin{cases} 1 \ if \ Yi = X \\ 0 \ otherwise \end{cases}$ with given random binary variables X, Yi. Obviously, *p(not X | Y1, Y2,…, Yn) = 1 - p(X | Y1, Y2,…, Yn).*

Applying formula 7, CPT of J, E, Q is determined below:

Table 1. CPTs of J, E, Q. Namely, T_1, T_2, T_3

T_1

C O I	p(J = 1)	P(J = 0) *1- p(J = 1)*
1 1 1	**1.0** *(0.1*1 + 0.5*1 + 0.4*1)*	**0.0**
1 1 0	**0.6** *(0.1*1 + 0.5*1 + 0.4*0)*	**0.4**
1 0 1	**0.5** *(0.1*1 + 0.5*0 + 0.4*1)*	**0.5**
1 0 0	**0.1** *(0.1*1 + 0.5*0 + 0.4*0)*	**0.9**
0 1 1	**0.9** *(0.1*0 + 0.5*1 + 0.4*1)*	**0.1**
0 1 0	**0.5** *(0.1*0 + 0.5*1 + 0.4*0)*	**0.5**
0 0 1	**0.4** *(0.1*0 + 0.5*0 + 0.4*1)*	**0.4**
0 0 0	**0.0** *(0.1*0 + 0.5*0 + 0.4*0)*	**1.0**

T_2

E	p(E = 1)	p(E=0) 1-p(E=1)
1	**0.8** (0.8*1)	**0.2**
0	**0.0** (0.8*0)	**1.0**

T_3

Q	p(Q = 1)	p(Q=0) 1-p(Q=1)
1	**0.2** (0.2*1)	**0.8**
0	**0.0** (0.8*0)	**1.0**

Because concepts C, O, I has no prerequisite knowledge for understanding, their CPTs are specified as prior probabilities obeying uniform distribution (assigned medium value 0.5 in most cases).

Table 2. CPTs of C, O, I. Namely, T_4, T_5, T_6

P(C=1)	P(C=0)
0.5	0.5

P(O=1)	P(O=0)
0.5	0.5

P(I=1)	P(I=0)
0.5	0.5

3.2 Inferring User's Knowledge

Suppose a leaner did well the exercise "Set up… " and asked more than 10 question. That is to say the occurrence of two evidence, namely, E = 1 and Q = 1. It is necessary to answer the question: *How mastered is learner over the concept "Java"*? Thus, the posterior conditional of hidden variables J with fixed events E =1 and Q=1, *p(J = 1 | C, O, I , E = 1, Q = 1)*, must be computed.

According to formula 5, 6:

$$p(J=1 \mid C,O,I,E=1,Q=1) = \frac{\sum_{C,O,I} p(J=1,C,O,I,E=1,Q=1)}{\sum_{C,O,I,E,Q} p(J=1,C,O,I,E,Q)}$$

where p(J, C, O, I, E, Q) is global joint probability distribution, p(J, C, O, I, E, Q) = p(C) * p(O) * p(I) * p(E|O) * p(Q|O) * p(J|C,O,I).

Applying all CPTs in table 1, 2, 3, 4, 5, 6, it is able to determined p(J, C, O, I, E, Q). After that, we compute p(J = 1 | C, O, I , E = 1, Q = 1) to answer above question.

Note, the set of all parents of a hidden node is the complete set of mutually exclusive hidden variables and the set of all evidence nodes which are children of a hidden node is the complete set of mutually exclusive evidence variables.

4 Conclusion

There is no doubt that the combination of Bayesian network and overlay model gives us the appropriate approach for user modeling but it has two disadvantages:

- The expense of data storage is high. A Bayesian network which has *n* variables together *n* CPTs with 2^n parameters (values in CPTs) under constraint: "each variable is binary (0 and 1)". If variables are not binary, the number of parameters are huge, so, it is difficult to store them in memory.
- The computation of posterior probability which is basis of inference consumes much time when executing in runtime because it is rather complex.

The first cause which is the inherent attribute of Bayesian network can be only restricted by programming technique when implementing network and it would be best to declare binary variables. On the other hand, it is the done to use CPT instead of continuous probability density/distribution function for solving the second problem.

As already discussed, the structure and parameters (CPTs) in our model are fixed and specified by experts. However, they must be evolved after each occurrence of

evidence. When learning machine is concerned, structural learning is process of gradual improving the structure of model and correspondingly, parametric learning is process of changing the parameters so as to be more suitable. We will discuss the improvement on qualitative model (structure) and quantitative model (parameters) in the other paper.

References

1. Akiba, T., Tanaka, H.: A Bayesian Approach for User Modeling in Dialogue Systems. In: COLING 1994. The 15th International Conference on Computational Linguistics, vol. 1 (1994)
2. Allen, R.B.: User Models: Theory, Method, and Practice. International Journal of Man-Machine Studies 32, 511–543 (1990)
3. Brachman, R.J., Levesque, H.J.: Knowledge Representation and Reasoning. ©2003 CMPT 411/882 Course Home Page (Spring 2005)
4. Bunt, A., Conati, C.: Probabilistic Student Modelling to Improve Exploratory Behaviour. Journal of User Modeling and User-Adapted Interaction 13(3), 269–309 (2003)
5. Conati, C.: Probabilistic Assessment of User's Emotions in Educational Games. Journal of Applied Artificial Intelligence, special issue on " Merging Cognition and Affect in HCI" 16(7-8), 555–575 (2002)
6. Charniak, E.: Bayesian Network without Tears. AI magazine (1991)
7. Chin, D.N.: A Case Study of Knowledge Representation in UC. In: Proceedings of the Eighth International Joint Conference on Artificial Intelligence, August 1983, vol. 1, pp. 388–390. Karlsruhe, West Germany (1983)
8. Chin, D.N.: KNOME: Modeling What the User Knows in UC. In: Kobsa, A., Wahlster, W. (eds.) User Models in Dialog Systems, pp. 74–107. Springer, Heidelberg
9. Fagin, R., Halpern, J.Y.: Reasoning about knowledge and probability. ACM 41(2), 340–367 (1994); Preliminary version appeared in: Vardi, M.Y. (ed.): Second Conf. on Theoretical Aspects of Reasoning about Knowledge, pp. 277-293. Morgan Kaufmann (1988); Corrigendum: J. ACM 45(1), p. 214 (January 1998)
10. Finin, T., Drager, D.: GUMS: A General User Modeling System. In: Proceedings of the Canadian Society for Computational Studies of Intelligence 1986 (CSCSI 1986) (1986)
11. Halpern, J.Y.: Reasoning about Knowledge, `http://www.cs.cornell.edu/home/halpern/topics.html#rak`
12. Halpern, J.Y.: Reasoning about Uncertainty, `http://www.cs.cornell.edu/home/halpern/topics.html#rau`
13. Henze, N., Nejdl, W.: Bayesian Modeling for Adaptive Hypermedia Systems. Knowledge Based Systems Group, University of Hannover, Lange Laube 3, 30159 Hannover, Germany (1999)
14. Horvitz, E., Breese, J., Heckerman, D., Hovel, D., Rommelse, K.: The Lumière Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI, July 1998, pp. 256–265. Morgan Kaufmann, San Francisco (1998)
15. Jameson, A.: Logic is not enough: Why reasoning about another person's beliefs is reasoning under uncertainty. In: Laux, A., Wansing, H. (eds.) Knowledge and Belief in Philosophy and Artificial Intelligence. Akademie Verlag, Berlin (1995)
16. Jameson, A., Hoeppner, W., Wahlster, W.: The Natural Language System HAM-RPM as a Hotel Manager: Some Representational Prerequisites. In: Wilhelm, R. (ed.) GI - 10. Jahrestagung Saarbrücken. Springer, Berlin (1980)

17. Jedlitschka, A., Althoff, K.-D.: Using Case-Based Reasoning for User Modeling in an Experience Management System. In: Proc. Workshop Adaptivität und Benutzermodellierung in Interaktiven Systemen (ABIS 2001), GI-Workshop-Woche Lernen - Lehren - Wissen - Adaptivität (LLWA 2001), Universität Dortmund, October 8-12 (2001)
18. Johansson, P.: User Modeling in Dialog Systems. St. Anna Report: SAR 02-2
19. Kobsa, A.: First experiences with the SB-ONE knowledge representation workbench in natural-language applications. ACM SIGART Bulletin 2(3), 70–76 (1991)
20. Kobsa, A.: User Modeling: Recent Work, Prospects and Hazards. Department of Computer Science, Columbia University, New York, USA (1993)
21. Orwant, J.L.: Doppelgänger Goes To School: Machine Learning for User Modeling. Master's thesis, Massachusetts Institute of Technology (1993)
22. Paiva, A., Self, J.: A Learner Model Reason Maintenance System. In: Cohn, A. (ed.) ECAI 1994. 11th European Conference on Artificial Intelligence. John Wiley & Sons, Ltd., Chichester (1994)
23. Papatheodorou, C.: Machine Learning in User Modeling. In: Paliouras, G., Karkaletsis, V., Spyropoulos, C.D. (eds.) ACAI 1999. LNCS (LNAI), vol. 2049, pp. 286–294. Springer, Heidelberg (2001)
24. Pohl, W.: Logic-Based Representation and Reasoning for User Modeling Shell Systems. In: User Modeling and User-Adapted Interaction (UMUAI 1999), vol. 9(3), pp. 217–283 (1999)
25. Pohl, W., Höhle, J.: Mechanisms for flexible representation and use of knowledge in user modeling shell systems. In: Jameson, A., Paris, C., Tasso, C. (eds.) User Modeling: Proceedings of the Sixth International Conference, UM 1997, CISM 1997. Springer, Vienna (1997)
26. Poole, D.: A Logical Framework for Default Reasoning. Logic Programming and Artificial Intelligence Group, Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, N2L3G1 (519), 888–4443 (1987)
27. Rich, E.: User Modeling via Stereotypes. Cognitive Science 3, 329–354 (1979)
28. Sparacino, F.: Sto(ry)chastics: a Bayesian Network Architecture for User Modeling and Computational Storytelling for Interactive Spaces. LNCS. Springer, Heidelberg (2003)
29. Ting, C.Y., Phon-Amnuaisuk, S., Chong, Y.K.: Modeling and Intervening Across Time in Scientific Inquiry Exploratory Learning Environment. Journal of Educational Technology & Society 11(3), 239–258 (2008)
30. Ting, C.Y., Reza Beik Zadeh, M.: Assessing Learner's Scientific Inquiry Skills Across Time: A Dynamic Bayesian Network Approach. In: Conati, C., McCoy, K., Paliouras, G. (eds.) UM 2007. LNCS, vol. 4511, pp. 207–216. Springer, Heidelberg (2007)
31. Tedesco, R., Dolog, P., Nejdl, W., Allert, H.: Distributed Bayesian Networks for User Modeling. In: ELEARN 2006: World Conference on E-Learning in Corporate, Government, Health Care, and Higher Education (2006)
32. wikipedia. Default Logic (2007), `http://en.wikipedia.org/wiki/Default_logic`
33. Wilensky, R.: Some Problems and Proposals for Knowledge Representation. Computer Science Division, University of California, Berkely, Report No. UCB/CSD 87/351 (1987)

Building Excitement, Experience and Expertise in Computational Science among Middle and High School Students

Patricia Jacobs

Shodor SUCCEED Apprenticeship Program
{patricia.jacobs}@shodor.org
http://www.shodor.org/succeed/apprenticeships

Abstract. Three of the most important skills for advancing modern mathematics and science are quantitative reasoning, computational thinking, and multi-scale modeling. The SUCCEED Apprenticeship program gives students the opportunity of exploring all three of these areas. The SUCCEED Apprenticeship program uses innovative approaches to get students excited about computational science. The overall goal of this program is to provide middle and high school students with authentic experiences in the techniques and tools of information technology with a particular focus on computational science. The program combines appropriate structure (classroom-style training and project-based work experience) with meaningful work content, giving students a wide variety of technical and communication skills. The program provides middle and high students from ethnically and economically diverse backgrounds with training and authentic experiences in using computational science.

Keywords: Computational Science, Math, Science, Technology, Engineering, Modeling, Interactive.

1 Introduction

Shodor, a national resource for computational science in Durham, N.C. is dedicated to the improvement of science and mathematics education by promoting the effective use of computer modeling and simulation technologies. The SUCCEED Apprenticeship program is one of a wide range of programs provided by Shodor. The overall goal of the SUCCEED Apprenticeship program is to provide activities and support mechanisms, and mentoring to move students from an excitement for computational science and Information Technology (IT) to becoming an expert in one or more areas of computational science and associated IT components

The SUCCEED Apprenticeship program provides students with authentic and appropriate experiences in the use of computational science and advanced technologies and techniques to study scientific events within the context of science, mathematics and engineering and to produce evidence that students become proficient in these skills. In addition to the computational and technical skills,

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 15–24, 2009.

the program also enable student apprentices to acquire a set of problem solving, collaboration and communication skills identified as valuable for 21st century workforce.

Significant work has proceeded during the program to develop and evaluate the project methodology of bridging the excitement-expert gap opportunities for upper middle and high school students in the local area.

2 Program Overview

The SUCCEED Apprenticeship program, which began in January 2006, builds on Shodors SUCCEED (Stimulating Understanding of Computational science through Collaboration, Exploration, Experiment and Discovery) program which provides workshops to introduce middle and high school students to technologies, techniques and tools of computational science. Once students have shown that excitement for math and science, they have the opportunity to participate in the SUCCEED Apprenticeship program. This program allows upper middle and high school students to work with Shodor staff and other scientists in a learning or apprentice mode. Apprentices use computational science to conduct scientific research, create mathematical models of scientific phenomenon and use those models to perform a variety of science and mathematical experiments.

The SUCCEED Apprenticeship program is targeted towards developing and evaluating activities and support mechanisms to move students from an excitement for computational science and IT, to becoming an expert in one or more areas of computational science and associated IT components. The project looks to primarily answer the question: what programs, activities, and support mechanisms are required to ensure that excited students become expert students in one or more areas, and how is that transition evaluated? The project methodology looks to bridge the excitement-expert gap by providing long-term, mentor-supported opportunities for upper middle and high school students in the local area.

During their participation in the program, apprentices take classes, work in project teams on local, regional and nationally funded projects, and have numerous opportunities to develop experience, culminating in the development of expertise in one or more areas of computational science. For example, apprentices may work in the field of computational chemistry, helping support Shodor's statewide computational chemistry computing services.

Through the combination of appropriate structure and meaningful work content, the SUCCEED Apprenticeship Program provides outstanding opportunities for students while providing the project staff with the mechanism by which to measure and evaluate this pipeline-building program.

3 Participants

The SUCCEED Apprenticeship Program, in its third year, has already surpassed its goal of working with 100 students over a three-year period. Participants are

upper middle and rising 9th-12th grade students who have an interest in science, technology, engineering and mathematics (STEM). Each apprentice is required to spend 780 hours in the program over the course of two years. Apprentices are recruited from Shodor SUCCEED workshops, local school-based programs and summer camp programs. Students are interviewed and admitted based on their interest in computational science, IT and STEM areas. Apprentices are paid a stipend over the course of their apprenticeships, which could be extended as long as two years.

During the Fall of 2007, the SUCCEED Apprenticeship program had approximately 60 applicants, twice as many applicants as we had space available for in the 2007-2008 class. Applicants were from diverse ethnic and socio-economic backgrounds. Each applicant completed an online application for consideration in the program. Each student was interviewed and evaluated in the following areas: their interest in the program, commitment to the program, teamwork, communication (both oral and written), and leadership.

After careful review of each applicant's application, and based on the comments from staff during the interviews, the program enrolled 50 new apprentices (beginner apprentices) for the 2007-2008 school year.

The new enrollment increased the total number of beginner and advanced apprentices enrolled in the SUCCEED Apprenticeship program for 2007-2008 school year to 71.

4 Program Structure and Curriculum

During the second and third year of the program, we continued to develop and evaluate the SUCCEED Apprenticeship program methodology. Apprentices were divided into cohort 1 (beginner apprentices) and cohort 2 (advanced apprentices) which depends on when they enrolled in the program. Due to the success and the number of students enrolled in the SUCCEED Apprenticeship program, the program has divided the students into different tracks (Track A and Track B). Students attend workshops depending on the Track in which they selected. Track A classes were held on the 1st and 3rd Saturday of each month, while Track B classes were held on the 2nd and 4th Saturdays each month. Both the advanced and beginner apprentices were divided into Tracks. Thus, classes were held every Saturday at Shodor for the beginner and advanced apprentices. The Track schedule allowed us to accommodate the large number of students in the program and to provide flexibility to the students.

Students in their second year of the program worked on different projects for local organizations. These projects challenged the knowledge of the apprentices and allowed them to work with real clients in the community. The first year apprentices completed and presented mini-projects on modeling, web design and other uses of innovative technology. In addition, some apprentices became apprenterns or completed the program to become interns at Shodor.

Throughout the year, all apprentices additionally improve math skills using computational tools; improve writing skills by completing weekly journals to

Program Structure and Curriculum

	School Year	Summer
Time commitment	▪ **18 hours per month**	▪ **6 weeks, full-time**
All Students Must	▪ **attend Saturday workshops twice a month** to learn new computational science and STEM skills ▪ spend additional **time working in the office** on projects	▪ attend **career days** and **team building activities**
Beginners (first year)	▪ attend workshops on **basic IT skills**: agent modeling, web design, programming, graphics ▪ **two-month subject modules** culminating in creative **group projects**	▪ **attend classes** in math, journaling, problem solving, creative thinking ▪ **help Shodor staff** on STEM projects
Advanced (second year)	▪ **teams** begin **projects for actual clients**: mentored by a Shodor staff members ▪ workshops on **software development process,** high level design, detailed design, database/ER diagrams, user interface, and testing	▪ **work on and complete client projects** ▪ projects must pass a quality assurance process of **verification and validation**

Fig. 1. SUCCEED Apprenticeship Program Structure

reflect on what they have learned and improve communication skills through regular discussions with their mentor and through oral presentation.

5 Program Activities and Findings

There are five major complementary and interdependent activities of the SUCCEED Apprenticeship program:

1. Teaching and supporting the appropriate and authentic use of IT-related technologies, techniques, and tools, with a particular focus on computational science and its associated areas.

Apprentices attended workshops to gain experience and develop expertise in one or more areas of computational science and associated uses of the technologies, techniques, and tools of IT, within the context of stem. Students were introduced to computational science through a two-week (60-hour), academically intensive education and research program. Students learned and used advanced computational science technologies, techniques, and tools to study a wide variety of scientific events. Topics involved general uses of computational science, basic numerical methods, scientific programming, model validation and verification and research methods incorporating computational science.

Throughout the 2007-2008 school year, the apprentices attended workshops, completed assignments and worked on group projects. Each apprentice had a time commitment of 18 hours a month at Shodor during the school year and 240 hours during the summer. The 18 hours a month consist of 2 classes and 2 hours a week in the afternoons. The apprentices attended workshops every other Saturday to learn new computational science and STEM skills. Outside of class, they were given assignments to practice and use new skills and group projects to demonstrate the knowledge of a given skill set.

Apprentices new to the program, beginner apprentices, had a more structured curriculum. Beginning apprentices attended classes in basic IT skills such as agent modeling, web design, programming and graphics. Their work was organized into modules of two to three month durations. After a module was completed, apprentices worked in teams to complete assigned projects. During the summer, the beginner apprentices attended classes in math, journaling, problem solving and creative thinking, held career days and team building activities.

Advanced apprentices (students in their 2nd year of the program) focused on a project based curriculum. All advanced apprentices were assigned to a team to work with mentors and clients on projects and to receive training as needed. Advanced apprentices learned advanced modeling, software development processes, high level design, detailed design, database/ER diagrams, graphics and quality assurance.

During Fall 2007, the Apprenticeship program added a new component, the Apprenticeship Math Component (AMC), to help students continue to improve their math skills. The AMC leverages the math resources in Shodor's award winning Interactivate courseware. Students are required to complete an associated set of activities in Number/Operations, Geometry/Measurement, Algebra and Probability/Statistics that are graded according to a rubric specific to the topic. Additionally, students are asked to write about the mathematics in the activity and assess the associated materials. Students are required to correct their work until it meets a particular score valued from the rubric.

All apprentices are required to re-work all of their assignments and projects until the quality of their work meet the rubrics or standards set by the staff and mentors. Once apprentices meet the requirements for class attendance, assignments and projects they receive a stipend. Stipend deadlines are set for four times throughout the year.

2. Provide mentors to define individual goals and timelines, and provide guidance through technical difficulties. Mentors monitor work progress as well as skill development for individual interns. In addition, mentors are responsible for overall research team dynamics,distribution of work and project oversight.

In addition to students attending classes, the SUCCEED Apprenticeship program continues to focus on mentoring students. The SUCCEED Apprenticeship program seeks to implement a 'true' apprenticeship program where young people learn from working with and learning from those with more experience. Shodor has approximately 18 staff who have a range of expertise in computational physics, biology, chemistry, math as well as computer science, system administration, graphics and web design. Each staff is assigned to mentor 6-7 students. Students are required to meet one hour per month with their mentor. Communication between mentors, program coordinator and parents is also on-going throughout the program.

In addition to monthly meetings, mentors track progress and skill development of apprentices by reviewing the apprentices weekly reflections (questions students have to answer weekly), progress on individual assignments and projects.

Many students receive additional mentoring from staff when they need help understanding and/or completing out of class assignments and projects. Staff also mentor and provide leadership for the advanced apprentices as they work in research teams on their projects. Staff help facilitate project plans, design and set up meetings with project clients.

3. Provide opportunities for apprentices to work on local, regional and nationally funded projects. The projects are done as a learning process, and thus will require intensive guidance to ensure quality workmanship.

Throughout the program, apprentices are provided with the opportunity to work on local, regional and nationally funded projects. During Fall 2007 and Spring 2008, we continued to partner with local organizations to provide real world experience for our advanced apprentices' projects. These projects range from working with with local organizations to learning and developing skills for Shodor's award winning resources such as CSERD (www.shodor.org/refdesk) and Interactivate projects (www.shodor.org/interactivate).

4. Providing instruction and opportunities to practice a wide variety of communication skills, including working effectively in a group, interacting with customers and clients, teaching younger students about the technologies and exercising leadership.

The SUCCEED Apprenticeship program provides opportunities for apprentices to practice a wide variety of communication skills, including working effectively in a group, interacting with clients as well as teaching STEM workshops for younger students. During 2007-2008, apprentices had to prepare presentations and present several projects they worked on to demonstrate their knowledge of the skills they learned. The presentations helped the apprentices improve their communication skills.

Apprentices were also given the opportunity to teach workshops about computational science technologies and tools to younger students at Shodor workshops. Apprentices taught workshops from system and agent modeling to system administration to middle and high school students.

5. Providing formal and informal opportunities in critical thinking, including data retrieval, data organization and analysis, application of evidence-based reasoning, problem-solving, creative thinking and decision making.

The SUCCEED Apprenticeship provides many opportunities for apprentices to use and demonstrate critical thinking skills. Apprentices must attend classes to learn computational science and STEM skills every other Saturday. Apprentices were given assignments to practice and use new skills. In addition, apprentices completed group projects to demonstrate a given skill set. Advanced apprentices worked in project teams to integrate various technical skills needed for the completion of their projects. These projects required that the apprentice use a variety of innovative skills as well as problem solving. Advanced apprentices learned and used skills such as modeling, PHP, MySQL, CSS, HTML, Javascript and Googlemap for their projects.

6 Evaluation

An extensive evaluation of the value added and the measurability of the Apprenticeship program is assessing the extent to which the appropriate structure and meaningful work content effectively develops students to become an expert in the areas of computational science and associated IT components. For 2007-2008, the SUCCEED Apprenticeship program had an overwhelming number of applicants for the program. For Fall 2007, we had a total number of 71 students enrolled in the program. We continued to evaluate the program's success and to improve the effectiveness of the program's structure and curriculum. Evaluation data combines students' participation, completion of assignments, journals, responses to routine surveys, as well as feedback from staff and others involved in the program.

In addition to working on their assignments and projects, apprentices are required to complete a web portfolio that displays their skills and knowledge learned at Shodor. Since the program began in 2005, an extensive evaluation process has helped us continually improve the effectiveness of the program's structure and curriculum. Not only have applications increased over the past three years, the percentage of students who stay with the program has nearly doubled. With the first group of program graduates, we see that these apprenticeships have been successful in maintaining apprentices' interest in science, math and technology as shown by their completion of the program and their career plans.

Goal 1: To build and maintain excitement for math and science in a diverse demographic of middle and high school students. The data presented below in Figure 2 show that the program has continued to attract and retain a diverse demographic of middle school students. Advanced Apprentices: In Fall 2007, 29 apprentices remained in the program through the summer of 2007. Twenty of these apprentices, then in their second year, returned for fall 2007. Four of these were accepted as Shodor Interns. Of those who returned the breakdown by gender and ethnicity was as follows: Female: 35%; African-American: 26%; Other Minorities: 26%. The largest subgroups were White Males and Minority Females. In addition, 14 apprentices continued in the program through Summer

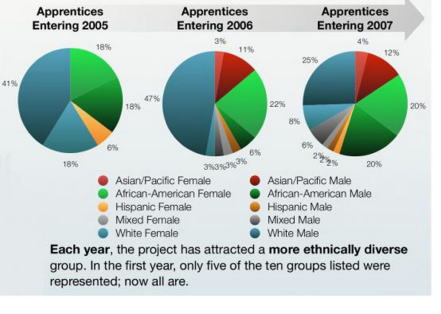


Fig. 2. SUCCEED Apprenticeship Demographics

Computer programming or engineering	13
Math or science	6
Medical	12
Other	8
Don't Know/No answer	8

Fig. 3. Apprentices Interest in STEM areas

of 2008 and completed advanced IT projects as described in a later section of this report. The breakdown by gender and ethnicity of this population was as follows: Female: 43%; Minority: 64%.

In a addition, to a diverse group of students, the SUCCEED Apprenticeship program also seeks to increase or maintain student's excitement for STEM. We have taken persistence in intention to pursue a career in STEM or a STEM-related career. as an indicator of excitement for STEM. Soon after being recruited in Fall 2007 beginning apprentices responded to a questionnaire that asked, among other items, to indicate their career goals.

GOAL 2. To enable student apprentices to acquire a set of technical, problem solving, collaboration and communication skills identified as valuable for the IT-intensive workplace.

This goal encompasses both computational science and technical (IT) skills and workplace (SCANS) skills which were assessed separately as follows.

Basic Skill Level - Technical Skills

The structure of the program, developed over the past three years, provides preliminary instruction followed by hands-on experience for all apprentices during the first year of the apprenticeship. At the end of the first year mentors rated beginning apprentices on a 1-5 scale on ability at the basic level to use the skills that had been taught. Mentors ratings are listed below as High(4 -5), Medium (3.0-3.9), Low (1-2.9).

Number of Apprentices at Each Level of Proficiency
Cohort 3, Year 1

N=35	High	Med)	Low	No Rating
JAVA	23	4	4	4
Source Control	20	7	4	4
UNIX	12	9	11	0
HTML/CSS	14	17	4	0
Graphics	6	28	1	0
PHP	0	35	0	0
NetLogo	24	5	2	5
EXCEL	19	5	3	8

Some apprentices have become more proficient than others but it is important to note that all have been introduced to these computer programs; those apprentices who are retained in the program will continue to improve these skills.

Fig. 4. Basic Skill Level

Number of Apprentices at Each Level of Proficiency
Cohort 2, Year 2

N=14				
	High	Med	Low	No Rating
Source Control	5	3	0	4
UNIX	4	8	2	
HTML/CSS	8	6	0	
Graphics	6	8	0	
PHP	8	4	2	
MYSQL	6	6	2	
EXCEL	14	0	0	
Obj. Orient. Prog.	7	7	0	

Fig. 5. Advanced Skill Level

Advanced Skill Level. Technical (IT) Skills
During the apprentices' second year these skills are developed further through individual and group projects that involve planning, execution, problem solving, reporting and presentation. As noted above, Advanced Apprentices were required to attend classes but were allowed more flexibility in hours since group projects are the main focus of the second year and successful completion of projects is the goal.

During the Summer session all Advanced apprentices participated actively in complex projects that required the use of IT skills well beyond the basic level. As in the previous year advanced apprentices were divided into four teams to work with mentors to design and produce interactive websites for clients in the community. At the end of the summer session each group prepared and presented a presentation to the sponsors of the project. Successful completion and acceptance by clients is clear evidence of the attainment of IT skills.

In addition to continuing informal assessment and feedback to apprentices, mentors rated the Advanced Apprentices (Cohort 2) at the end of their second year on a scale of 1-5 on a range of IT skills. For this rating the criteria were more stringent than on the rating after the first year. Those given a rating of 4-5 were deemed expert and able to teach the skill to others. A rating of 3 indicated successful use of the skill in projects assigned and completed. Although a rating of 1 was possible, none of the apprentices received this rating, indicating that all apprentices had developed at least a minimum level of proficiency in every skill.

The results of the assessment by mentors who have been actively engaged with apprentices indicates that, with very few exceptions, all apprentices who persisted in the program are able to use the IT skills listed above at an expert or practical level in any environment where these skills are required.

GOAL 4: To gather evidence of the effectiveness of this structure for increasing participation of traditionally underrepresented demographic groups in IT-intensive workplaces.

The Apprenticeship program, as it has developed over three years, enables students of diverse backgrounds to gain IT proficiency. We have shown that students of diverse racial and ethnic backgrounds have gained proficiency in a wide range of IT skills. T he percentage of underrepresented groups remains high among the apprentices who persist through the entire 26-week program. In addition, many students who have participated in the program expect to have IT-related careers and thus will take their places in IT-intensive workplaces as they move into the adult workforce. An interesting and unusual aspect of the population of apprentices in the program is the high percentage of African American females. We believe that interaction between students of diverse races, ethnicities and genders in a learning environment where respect for all students, as well as mentors and instructors, is the norm and is expected and required has been an important aspect of the program.

The results of the assessment by mentors who have been actively engaged with apprentices indicates that, with very few exceptions, all apprentices who persisted in the program are able to use the IT skills listed above at an expert or practical level in any environment where these skills are required.

7 Summary

The first two of three groups (cohorts) of Apprentices have now completed the 2-year Apprenticeship program and a third group is continuing through the current year. Since enrollment in the program began in late 2005 a total of 124 students have been admitted into the program; 86 have completed the first year of the program and 29 have completed the full 26-week program. Approximate 20 additional apprentices are expected to complete the full program in Summer 2009.

For three successive years the project has been notably successful in recruiting a diverse group of students and in maintaining diversity in the groups who have persisted through the full program. The program has also been successful in maintaining apprentices' interest in STEM and STEM-related careers. The program, now entering its final year, is expected to continue the curriculum and other practices that have been tested and found to be effective over the past three years. The curriculum, lesson plans and resources are being made available for others interested in developing a similar program at (http://shodor.org/succeed/curriculum/apprenticeship/).

Using R for Computer Simulation and Data Analysis in Biochemistry, Molecular Biology, and Biophysics

Victor A. Bloomfield

Department of Biochemistry, Molecular Biology, and Biophysics
University of Minnesota
321 Church St. SE
Minneapolis, Minnesota 55455
victor@umn.edu

Abstract. Modern biology has become a much more quantitative science, so there is a need to teach a quantitative approach to students. I have developed a course that teaches students some approaches to constructing computational models of biological mechanisms, both deterministic and with some elements of randomness; learning how concepts of probability can help to understand important features of DNA sequences; and applying a useful set of statistical methods to analysis of experimental data. The free, open-source, cross-platform program R serves well as the computer tool for the course, because of its high-level capabilities, excellent graphics, superb statistical capabilities, extensive contributed packages, and active development in bioinformatics.

1 Introduction

1.1 The Need for a More Quantitative Bbiology

The Executive Summary of the influential 2003 report from the National Academy of Sciences, "BIO 2010: Transforming Undergraduate Education for Future Research Biologists" [1], begins

> The interplay of the recombinant DNA, instrumentation, and digital revolutions has profoundly transformed biological research. The confluence of these three innovations has led to important discoveries, such as the mapping of the human genome. How biologists design, perform, and analyze experiments is changing swiftly. Biological concepts and models are becoming more quantitative, and biological research has become critically dependent on concepts and methods drawn from other scientific disciplines. The connections between the biological sciences and the physical sciences, mathematics, and computer science are rapidly becoming deeper and more extensive.

Quantitative approaches have become particularly prominent in the large-scale approaches of systems biology and its associated high-throughput techniques: bioinformatics, genomics, proteomics, metabolomics, cellomics, etc.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 25–33, 2009.

High levels of quantitation are also needed in some of the more biophysically-oriented aspects of biochemistry, molecular and cellular biology, physiology, pharmacology, and neuroscience.

The increasing use of quantitation at the frontiers of modern biology requires that students learn some basic quantitative methods at an early stage, so that they can build on them as their careers develop. To deal with realistic biological problems, these quantitative methods need to go beyond those taught in standard courses in calculus and the elements of differential equations and linear algebra—courses based mainly on analytical approaches—to encompass appropriate numerical and computational techniques. The types of realistic biological problems that contemporary science is facing are generally too large and complex to yield to analytical approaches, and specific numerical answers are usually desired, so in many cases it makes sense to go directly to computational rather than analytical mathematical answers.

Modern molecular and cellular biology also demands increasingly sophisticated use of statistics, a demand difficult to meet when many life science students don't take even an elementary statistics course.

1.2 Computer vs. Analytical Tools

To add significant instruction in computational and statistical methods to an already overcrowded biology curriculum poses a challenge. Fortunately, modern computer tools, running on ordinary personal computers, enable very sophisticated analyses without requiring much analytical or programming knowledge or effort. In essence, this is a "black box" approach to quantitative biology; but I contend that using a set of black boxes is better than not using quantitative tools at all when they would substantially enhance the results of biological investigations. The challenge, then, is to make students—and more mature scientists—aware of the appropriate black boxes, their capabilities, and the steps needed to access those capabilities. With modern computer tools, this requires only a small amount of programming and an even smaller amount of analytical manipulation.

1.3 The Choice of R as the Computational Tool

R is a free software environment for computer programming, statistical computing, and graphics. The R web site [2] emphasizes statistical computing and graphics, which it does superlatively well; but R is also a very capable environment for general numerical computer programming.

R has many characteristics that make it a good choice on which to build quantitative expertise in the biochemical sciences. Its capabilities are similar to those of excellent and widely-used but expensive commercial programs. It runs on Mac OS, Windows, and various Linux and Unix platforms. It is free, open-source, and undergoing continual (but not excessive) development and maintenance. It is an evolving but stable platform that will remain reliable for many years. It has a wide variety of useful built-in functions and packages, and can be readily extended with standard programming techniques. It has excellent graphics.

If needed for large, computationally demanding projects, R can be used to interface with other, speedier but less convenient programming languages. Once its (fairly simple) syntax is learned, it is easier and more efficient than a spreadsheet. It has many sample datasets, which help with learning to use the program. It is widely used in statistics, and is increasingly used in biological applications, most notably the Bioconductor project [3].

Because of these characteristics, R can serve students as their basic quantitative, statistical, and graphics tool as they develop their careers.

2 Syllabus for the Course

The course is designed for one semester. It is divided into four main parts, with 14 "modules" corresponding to 14 weeks of the course and chapters of the accompanying textbook [4]. Two additional weeks are allocated for midterm and final examinations.

2.1 Part 1: The Basics of R

Calculating. In this and the next module, we begin with the most basic aspects of R: installing it, checking the installation by demonstrating some of its impressive graphics, and showing how it can be used as a powerful calculator with vector and matrix capabilities.

Plotting. An important part of scientific computing and data analysis is graphical visualization, an area in which R is very strong. R has many specialized graph types, some of which are explored later in the course. However, for many scientific purposes just a few types will suffice, especially graphs for data, functions, and histograms. We first give simple examples, and then show how they can be customized.

Built-in functions, user-defined functions, and programming. The base installation of R has many built-in functions, including `sort` and `order` to arrange vectors or arrays in ascending or descending order; all the standard trigonometric and hyperbolic functions `log`(base e), `log10`, `exp`, `sqrt`, `abs`, etc.; and more sophisticated mathematical functions such as `factorial`, `gamma`, `bessel`, `fft` (Fourier transform), etc. Additional mathematical functions, the orthogonal polynomials used in mathematical physics and chemistry, are available in the contributed package `orthopolynom`, available through the CRAN web site [2]. The functions `uniroot` and `polyroot` are used to solve for the zeros of general functions and polynomials, respectively. In addition to those mathematical functions, R has numerous others that are useful to scientists, including sorting, splines, and sampling. We also show how to define new functions, with examples of Gaussian functions and pH-titration curves.

Programs in R, as in most computer languages, typically consist of a few standard types of operations: assigning the values of variables and evaluating expressions involving those variables; conditional execution, in which different

sequences of statements are executed depending on whether a expression is true or false; and repetition or looping, in which an action is performed repeatedly until some condition is met. R is generally thought of as a programming language for statisticians, but it has the capabilities needed for the sort of numerical analysis done in most sorts of scientific work. The module concludes with some of the most common examples: finding the roots of polynomials or other functions, solving systems of linear and nonlinear equations, and numerical integration and differentiation.

Other important tasks, such as numerically solving differential equations, fitting data to linear or nonlinear equations, and finding periodicities in data with spectral analysis and Fourier transforms, are introduced in subsequent chapters.

Data and packages. Up to this point we have mainly dealt with how to use R for calculating and graphing. In programming for scientific work we also generally need to get data from various sources, transform it, and save it for later use. We also will often wish to augment the built-in capabilities of R with more specialized resources. Many such resources are available as contributed packages from the CRAN web site [2]. This module deals with those two important topics: handling data and adding packages.

2.2 Part 2: Simulation of Biological Processes

Equilibrium and steady state calculations. Much of biochemistry, molecular biology, and biophysics deals with the the equilibrium and dynamics of biochemical reactions. In this module we focus on two important types of time-independent processes: ligand binding and steady-state enzyme kinetics. These serve as test beds for showing how to use the plotting and data analysis capabilities of R.

Differential equations and reaction kinetics. In this module we show how to numerically solve the kinetic rate equations of the sort that describe biochemical metabolism, microbial growth, and similar biological phenomena. These systems of ordinary differential equations describe the change of concentrations or numbers of organisms as a function of time.

Population dynamics: competition, predation, and infection. Populat ions—whether of organisms, cells, or viruses—are of central importance in biology. In this module we consider some of the basic models of population dynamics: competition of different species for resources, predation of one species upon another, and transitions of parts of a population between different states (e.g., susceptible, infected, resistant, dead) in epidemics.

Diffusion and transport. The movement of biological molecules in cells or in lab experiments gives useful insight into their sizes, associations, and mechanisms of transport to functional locations. The movement may be random diffusion (Brownian motion), it may be in response to some driving force, or both. Familiar driving forces in the lab are electrophoretic and centrifugal fields. In the

cell, active transport and transport by cytoskeletal fibers are important mechanisms. Related situations arise in drug delivery, where the flow of drug from one compartment of the body to another can be treated by diffusion and transport models. Diffusion may be coupled with reaction as discussed in a section on regulation of morphogenesis. In this module we develop simple simulations for some of these processes. These simulations involve an introduction to the solution of partial differential equations with both space and time as independent variables.

Regulation and control of metabolism. Metabolism involves not just single biochemical reactions, but coordinated networks of reactions. These networks are usually remarkably well-regulated, keeping close to a set-point, a steady state or dynamic equilibrium in most healthy organisms. Substantial deviation from that set-point may betoken disease or some other extraordinary circumstance. On the other hand, biotechnologists may want to manipulate an organism to overproduce a desirable product, controlling its metabolism to deviate from the normal set-point. In this module we examine these issues of regulation and control by simulating the behavior of networks of enzymatic reactions.

Models of regulation. This module considers models of regulation in three different types of biological processes: transcription, response to chemotactic signals, and patterning of morphogens in cellular development. These are each huge topics, and we attempt only to present some introductory but instructive examples that are amenable to numerical simulation. An important theme is *robustness*, the ability of a system to maintain suitable functioning in the face of variations, both temporal and cell-to-cell, of biochemical parameters.

2.3 Part 3: Probability and Sequence Analysis

Probability and population genetics. Up to this point we have mainly treated deterministic processes, although we have showed how to fit noisy data and to model stochastic chemical reactions. In fact, most biological data are intrinsically noisy, or random, due to the underlying nature of the process (e.g., mutation or genetic recombination), especially when combined with the often small numbers of "individuals" in many experiments. This module discusses basic concepts of randomness and probability, and shows how these concepts may be applied in a variety of situations, concluding with a brief introduction to population genetics.

DNA sequence analysis. In this module we introduce some of the elementary concepts for analyzing DNA sequences in terms of "words" of length 1 (bases), 2 (base pairs), 3 (triplets, such as codons), restriction sites, etc. The analysis uses the basic probability concepts from the previous module.

2.4 Part 4: Statistical Analysis in Molecular and Cellular Biology

Statistical Analysis of Data. Molecular biologists and biophysicists have a lot of data to analyze, and R has a lot of tools to help with the analysis.

We consider three major topics: summary statistics for a single group of data, statistical comparison of two samples, and analysis of spectral data.

Microarrays. DNA microarrays are one of the key new technologies in biology. They are used to measure changes in gene expression levels, to detect single nucleotide polymorphisms (SNPs) that may be indicators of susceptibility to disease or useful in forensic analysis, to compare genome content of different cells or closely related organisms, and to detect alternative splicing in DNA transcription. A microarray may contain ten thousand or more spots, and therefore can carry out thousands of comparative genetic analyses at once. This enormous amount of information can provide great insight into genetic regulatory processes, but it also poses great challenges to data quality and adequate statistical analysis. This module provides a brief introduction to these issues.

3 Experience Teaching the Course

3.1 Learning and Using R

At the beginning of the course, students are told to download and install R from the CRAN (Comprehensive R Archive Network) web site [2]. To my pleasant surprise, in two offerings of the course to 28 students, none has had any trouble installing R regardless of their operating system (Mac OS, Windows, or Linux). An immediate demonstration of some of R's capabilities is obtained by running the graphics demonstration `demo(graphics)`.

The syntax of R is relatively simple and students have little trouble with the basics. The R program, as installed on the students' computers, has an extensive Help facility accessed from the menu bar. "An Introduction to R" and "R Data Import/Export" are likely to be useful as they begin learning the language. Each function has a help page, with definitions of the inputs, outputs, and options, and one or more examples of usage. These examples, however, are often terse and technical rather than readily tutorial.

A problem with the R help system is that you generally have to know the exact term being searched for, since the help system searches a pre-established index rather than the full text. For example, trying to learn about correlation analysis by typing `help(correlation)` or `?correlation` yields "Help topic not found". `?corr` gives the same result. Finally, `"cor"` brings up the desired help page. The functions `apropos` and `help.search` may (or may not) be useful in such cases. Two aids to finding online help about R topics are *RSeek* [5] and *Search the R Statistical Language* [6].

There are numerous online sites devoted to R. A particularly useful one for rapid reference is *R & BioConductor Manual* by Girke [7]. Another handy resource is the on-line *R Reference Card* by Short [8].

3.2 Useful Books

Most of the books that teach how to use R (or its progenitor S or commercial sibling S+) do so in the context of its use as a program for doing statistics.

Statistics is only one of the foci of the course, but the books by Dalgaard [9] and Verzani [10] provide useful introductions to R.

More advanced books that use R in a biological context, especially in bioinformatics, include those by Deonier et al [11], Gentleman [12], Paradis [13], Gentleman et al [14], and Hahne et al [15]. The book *Stochastic Modelling for Systems Biology* by Wilkinson [16] uses some R code in its treatment of systems biology.

In my development of this course, I have drawn heavily on *Computer Simulation in Biology: A BASIC Introduction*, by R.E. Keen and J.D. Spain (1992) [17]. This book, which appears to be out of print, uses BASIC, an earlier and much less capable computer language than R; but it has a good selection of topics and computer simulation examples for an introductory course. Of the many recent books on mathematical and computational biology, the two that fall closest to my approach, in their selection of topics and in emphasizing computational rather than analytical approaches, are those by Fall et al [18] and Allman and Rhodes [19].

3.3 Student Reaction

The course is intended for advanced undergraduates and beginning graduate students who have had basic instruction in biochemistry and calculus-level mathematics. In the two offerings thus far, the students have been a mix of senior undergraduates (most majoring in biochemistry) and beginning graduate students (the majority in masters programs in biology or microbial engineering). A few others have come from other disciplines such as computer science and chemical engineering. Neither group is very strong in analytical mathematics beyond basic calculus and linear algebra.

This diversity of backgrounds means that some students are stronger in the biological sciences, and others in mathematics and computers. Students who have had previous programming experience, but little biology, have performed better than those with a lot of biology but not much computer background. The biology students who have had the most difficulty are those whose quantitative backgrounds and interests are not strong.

Preparation in basic quantitative biochemistry (pH, equilibrium, reaction kinetics) is not strong, despite prerequisites. A number of students had particular trouble with chemical equilibrium calculations, material to which they should have been exposed in several previous courses. However, this material is notoriously difficult for mathematically-challenged students, a description that applies to many life science majors. It also appears that some of the practical implementation of equilibrium ideas, such as using difference spectroscopy to measure the relative amounts of species in a reacting mixture, are not adequately taught in prerequisite courses.

In a midterm evaluation, most of the students indicated they had a good understanding of what was expected of them, and that the combination of online lecture notes (the chapters of the book) and comments through the listserv were adequately clear. Some would have liked more illustrations or examples.

There was a diversity of opinion about whether the course got them interested or involved, perhaps because it was a required course for most of the graduate students. Nearly all agreed that the class required considerably more work than similar classes they have taken. It probably should be changed from a three-credit to a four-credit course.

Most of the students seemed to learn a lot, and could do fairly sophisticated problems by the end of the course. A few students never seemed to develop the facility, however. Part of this may have been my assumption of too much prior knowledge, so that theses students became overwhelmed and never caught up. In addition, most biology students are not used to the idea that they need to get things exactly right, otherwise the code won't work.

These difficulties may have been exacerbated by the fact that this has been taught as an on-line course, though under other circumstances it could certainly be taught in a regular classroom setting. Some students have indicated that they would like regular face-to-face sessions, but arranging the timing has proven difficult. A listserv makes asking and answering questions prompt and straightforward, and the students help each other both on the listserv and in group study sessions; but some of the explanatory and motivational things that typically go on in class may get shortchanged in the on-line format.

4 Conclusion

Overall, the course has been successful in teaching students from a variety of disciplines to use computational methods to model and analyze biological phenomena. R is an excellent computational tool for this purpose. The course covers a wide range of pertinent topics in biochemistry, molecular biology, and biophysics, many at a level thaat could not be adequately handled without computational tools. Other instructors could readily modify this list of topics to meet local needs and interests. Students who master this material are well-prepared to use high-level computational approaches to modern biology as their careers progress.

References

1. National Research Council: BIO 2010: Transforming Undergraduate Education for Future Research Biologists (2003)
2. The Comprehensive R Archive Network, `http://cran.r-project.org/`
3. The Bioconductor Project, `http://www.bioconductor.org/`
4. Bloomfield, V.A.: Computer Simulation and Data Analysis in Molecular Biology and Biophysics: An Introduction Using R. Springer, Heidelberg (2010) (in press)
5. `http://www.rseek.org/`
6. Search the R Statistical Language, `http://www.dangoldstein.com/search_r.html`
7. Girke, T.: R & Bioconductor Manual, `http://faculty.ucr.edu/tgirke/Documents/R_BioCond/R_BioCondManual.html`
8. Short, T.: R Reference Card, `http://cran.r-project.org/doc/contrib/Short-refcard.pdf`

9. Dalgaard, P.: Introductory Statistics with R. Springer, Heidelberg (2002)
10. Verzani, J.: Using R for Introductory Statistics. Chapman & Hall/CRC, Boca Raton (2005)
11. Deonier, R.C., Tavaré, S., Waterman, M.S.: Computational Genome Analysis: An Introduction. Springer, Heidelberg (2005)
12. Gentleman, R.: R Programming for Bioinformatics. Chapman & Hall/CRC, Boca Raton (2008)
13. Paradis, E.: Analysis of Phylogenetics and Evolution with R. Springer, Heidelberg (2006)
14. Gentleman, R., Carey, V.J., Huber, W., Irizarry, R.A., Dudoit, S. (eds.): Bioinformatics and Computational Biology Solutions Using R and Bioconductor. Springer, Heidelberg (2005)
15. Hahne, F., Huber, W., Gentleman, R., Falcon, S.: Bioconductor Case Studies (Use R). Springer, Heidelberg (2008)
16. Wilkinson, D.J.: Stochastic Modelling for Systems Biology. Chapman & Hall/CRC, Boca Raton (2006)
17. Keen, R.E., Spain, J.D.: Computer Simulation in Biology: A BASIC Introduction. Wiley-Liss, Chichester (1992)
18. Fall, C., Marland, E., Wagner, J., Tyson, J. (eds.): Computational Cell Biology. Springer, Heidelberg (2002)
19. Allman, E.S., Rhodes, J.A.: Mathematical Models in Biology: An Introduction. Cambridge University Press, Cambridge (2004)

Teaching Model for Computational Science and Engineering Programme*

Hayden Stainsby, Ronal Muresano, Leonardo Fialho, Juan Carlos González, Dolores Rexachs, and Emilio Luque

Universitat Autònoma de Barcelona
Computer Architecture and Operating System Department (CAOS)
Barcelona, Spain
{hstainsby,rmuresano,lfialho,jgonzalez}@caos.uab.es
{dolores.rexachs,emilio.luque}@uab.es

Abstract. Computational Science and Engineering is an inherently multidisciplinary field, the increasingly important partner of theory and experimentation in 0the development of knowledge. The Computer Architecture and Operating Systems department of the Universitat Autònoma de Barcelona has created a new innovative masters degree programme with the aim of introducing students to core concepts in this field such as large scale simulation and high performance computing. An innovative course model allows students without a computational science background to enter this arena. Students from different fields have already completed the first edition of the new course and positive feedback has been received from students and professors alike. The second edition is in development.

1 Introduction

Computational Science and Engineering (CSE) is the increasingly important partner of theory and experimentation in the development of knowledge. CSE requires multidisciplinary work, which allows the undertaking of complex scientific and engineering problems, under the unifying concepts of computing and mathematics.

Different industrial sectors such as medicine, life sciences, mechanics, economy, social sciences and management together with engineering use key CSE techniques, like modelling and simulation, to aid in solving their problems [1].

Due to this requirement the Computer Architecture and Operating Systems (CAOS) department of the Universitat Autònoma de Barcelona (UAB) initiated an innovative masters programme in Computational Science and Engineering. The members of the CAOS department have previous experience in these areas, for some years they have been performing research in the fields of advanced parallel and distributed simulation techniques, such as individual orientated simulation in biology and forest fire propagation, using High Performance Computing.

* Supported by the MEC-Spain under contract TIN2007-64974.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 34–43, 2009.

The results of this experience have been a number of masters and doctoral (PhD) theses as well as research papers and participation in different research projects funded by the European Union.

The programme has been designed according to the European Higher Education Area (EHEA), the so-called Bologna process, the current European standardised university degree system [2].

The masters programme in CSE is geared towards accepting students of various backgrounds across different sectors. This includes students from all areas that make use of mathematical modelling and scientific computing technologies.

Further to the core subjects, depending on the student's background, a different set of subjects will be offered to complement their knowledge according to the skills expected at the conclusion of the programme.

Upon concluding the programme, students will have the capability to identify computational needs in the areas in which they are working. The degree gives students the option of following either an academic or professional career path.

In this programme students will be given opportunities to join and perform within a research group as well as the possibility of working with high performance computers. Students learn investigation techniques, experience participating in multidisciplinary groups, and working with other people with expertise in different areas. These skills are applied in the development of the masters thesis for students following a research specialisation, or project report for the students focussing on professional skills.

This paper is written as follows. Section 2 discusses the course model, both the process of knowledge standardisation and the core and optional subjects. Section 3 presents the experiences obtained and observations made after the first year of running the CSE masters programme, shown using examples of work from masters theses. Finally, the conclusions are stated in section 4.

2 Overview of the Proposed Course Model

The classical concept of three pillars of knowledge, upon which computational science is based, has driven the model for the masters degree in Computational Science and Engineering. These three pillars are Mathematics, Computation, and Science and Engineering as shown in Figure 1. The relationship between these three pillars and the content of the CSE masters programme is discussed in section 2.2.

According to the Bologna Process to achieve a masters degree a student must have completed a total of 300 European Credit Transfer System (ECTS) credits including the bachelor degree (180 ECTS) or graduate (240 ECTS) credits. This model accepts students who have already completed more than 180 ECTS credits [3][4].

The core subjects of this masters programme are worth a total of 60 ECTS credits. The programme is divided in three compulsory subjects and one optional subject each worth 10 ECTS credits and a project which is worth 20 ECTS credits. There are nine additional subjects, each also worth 10 ECTS credits. The complete course structure is shown in Figure 2.

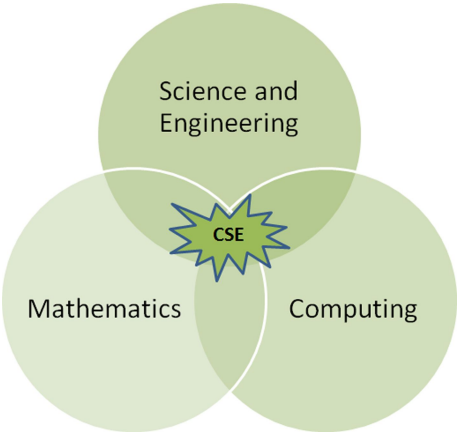


Fig. 1. Three pillars of Computational Science and Engineering

Students who have already completed 240 ECTS credits and have the requisite computational science background are able to begin the masters course with the core subjects, the second year of the programme. Other students may need to complete all or part of the first year. There are two possible reasons for this, the first is that the student has only completed 180 ECTS credits, usually analogous to three years of study under the ECTS. The second reason is that the student requires additional knowledge to follow the core of the course, this is discussed further in Section 2.1.

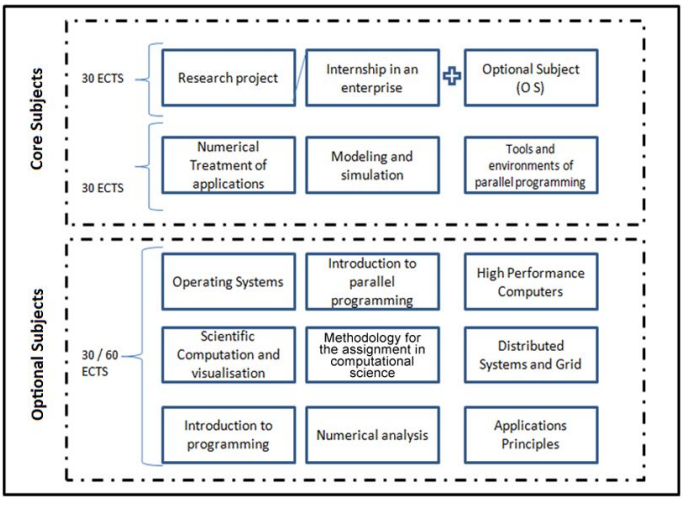


Fig. 2. CSE Course Structure

2.1 Knowledge Standardisation

This Computational Science and Engineering masters programme is designed to accept students from various academic fields. A challenge of this model is that students may begin the course without sufficient background in one of the fundamental areas of the course, such as computational science, mathematics, or parallel applications. In order to solve this problem, the masters programme has proposed a criteria to help standardise the students' knowledge in computational science.

The proposed criteria is based on two phases. The first phase starts with the analysis of each candidate's previous academic experience, sent during their application. In the second phase, once the candidate has been accepted, they attend an interview with the course director in order to define what additional or complimentary knowledge they will require. The optional subjects of the masters programme will provide the opportunity for the student to gain this knowledge.

A multidisciplinary committee will analyse each candidate's academic experience in order to define which subjects from the student's previous studies could be credited towards the masters course model. Taking into account the number of ECTS credits which the candidate must achieve to complete a masters degree, a different number of subjects may be required. This analysis takes into account the areas in which the candidate has previous academic experience, e.g. a candidate who already has some programming experience will most likely be better equipped to understand the course core than another candidate with no equivalent experience. There are a number of different aspects that are taken into consideration when choosing the optional subjects which will make up the student's knowledge standardisation. Some examples are listed in Table 1.

The possibility of choosing these optional subjects permits the student to compose a more personalised programme. The completion of the optional subjects also allows the student to follow the course core.

2.2 Core Subjects

The core of the course is comprised of two parts. The first part involves three core subjects and forms the basis of the skills students will learn in the masters programme. The second part allows students to orientate themselves towards research or professional skills, by choosing either the masters thesis or a professional internship.

While each of the core subjects features aspects of one of the three pillars more than the others, just like Computational Science and Engineering itself they are composed of all three key areas. Tools and Environments for Parallel Programming draws mainly from Computing , Modelling and Simulation is seated mostly within Science and Engineering, and Numerical Treatment of Applications is based in Mathematics.

Tools and Environments for Parallel Programming. Focusses on introducing students to parallel programming and the variety of tools available to

Table 1. Important aspects of a student's previous experience and examples

Academic experience	The specific topics the student has studied in previous academic courses. This also takes into account additional certificates and accreditations in relevant fields, e.g. an economist who has completed a course in financial mathematics may not need to take a numerical analysis subject.
Professional experience	Particular aspects of the student's professional career which are relevant to the CSE course, e.g. an electronic engineer who has worked with micro-controller programming will most likely not need to take the introductory programming subject.
Masters orientation	According to the student's motivation for starting the masters programme, whether they are considering a professional specialisation or to begin a research career, e.g. a student wishing to follow a research path would find subjects such as Methodology for the assignment in computational science or Scientific computation and visualisation of most interest.
Area of interest	Corresponding to the field in which the student wishes to apply the knowledge gained in this masters course, different subjects should be considered, e.g. a biologist who wishes to work in the management of a research centre which makes use of large scale calculations would attend subjects oriented towards computer architecture such as High performance computers or Distributed systems and GRID.

assist in all stages of the design, construction, and analysis of parallel applications. In this subject the students must make use of knowledge from all areas of CSE.

As part of this subject, the student will choose a problem to solve through the use of parallel computing. The subject is structured around providing the student with the skills and knowledge necessary to complete this project. The course begins with introductions to various methods of parallel programming, such as message passing and shared memory. During this time the student will select a topic and present it, along with how they propose to create a parallel version, to the class. In this presentation feedback is given, both by the class instructor and fellow students. Using this knowledge, the student then goes away to complete the parallelisation and performance analysis of their project. At the same time the class topics begin to cover a range of skills that will assist the student in this task, skills such as performance modelling and analysis, and scalability.

While the student is working on their personal project they may encounter difficulties, whereupon they are encouraged to seek assistance from their class instructor. This interaction provides a more personalised approach to teaching,

Table 2. Course content for Tools and Environments for Parallel Programming

Parallel programming	Message passing, shared memory, and skeletons
Parallel paradigms	Master-worker, pipeline, and single programme multiple data
Performance analysis	Performance models, dynamic instrumentation, and automatic tuning tools

where the instructor can give advice on tools and methods relating to the student's specific problem field.

At the end of the subject, the student will give a second presentation to the class detailing the process they went through in creating a parallel version of their problem and the performance that it has achieved.

Throughout this entire process the student will learn far more than how to write parallel code. They will understand the processes involved in analysing a serial algorithm and making efficient use of resources in order to parallelise it.

Table 2 shows the subject contents.

Modelling and Simulation. Aims to develop a model which represents a problem in a specific research field. This permits the student to apply methodologies, following scientific criteria, in order to define and extract conclusions and evaluate performance on the proposed problem.

The student first learns the basics of modelling, the considerations that need to be made when a real world system is modelled and how the expected outcome of a model effects decisions about which aspects of a system are important in the model.

This knowledge is used when the student goes on to learn about the various methods of simulation, especially the uses of system dynamics and discrete event simulation techniques. This leads into the importance of the analysis of simulation data and design practices.

Throughout a number of classes in this subject specific simulation areas are presented by experts in their respective fields. The student will understand how modelling and simulation is applied in real world situations, for example classes are given by a sociologist with computational experience. The student is shown a specific social sciences model and the method used to solve it using simulation software. This hands-on experience teaches the student the techniques and tools used to solve many kinds of problems. Other use cases give the student the appreciation of specific aspects of simulation.

During this subject the student will understand not only the basic concepts of how to develop models of real world systems and simulate them using computer software, but ways in which this has been applied to specific problem areas.

Table 3 shows the subject contents.

Numerical Treatment of Applications. Teaches students to apply the numerical techniques of computational modelling for applications in specific fields

Table 3. Course content for Modelling and Simulation

Modelling	Types of models (Heuristic and Empirical), application field (Conceptual and mathematical), representation field (Qualitative and Numerical) and verification and validation
Simulation	Physical systems and simulation, language and tools, performance metrics and distributed high performance simulation
Application use cases	Biological, economic and social systems, and propagation of forest fires

of science and engineering. This subject is a link between Modelling and Simulation and Tools and Environments for Parallel Programming just as mathematics is the path between a problem and the problem solving process.

In the first part of the subject the student learns mathematical techniques and how they are applied in a computing system. This covers such topics as floating-point errors, data mining, and efficient methods for storing data. This is important when explaining the solving of linear equation systems, where the student will use different solving methods and compare them using mathematical software.

The student will choose a topic in the area of statistical modelling which they will present to the class. This builds upon what the students have already been taught in class and allows the student to investigate a topic of interest to them in more depth. Another presentation is given by the student on different techniques for solving differential and integral calculus. These techniques are analysed and their potential for parallelisation is investigated.

This method of constant student feedback in the form of projects and presentations given to their peers allows a degree of interaction between the students and the instructor that gives an insight into these topics with the objective of identifying and reinforcing the key points.

The subject contents are shown in Table 4.

Table 4. Course content for Numerical Treatment of Applications

Mathematics	Application to problem modelling, linear equation systems, statistics, and integral and differential calculus
Application	Efficient methods of storing information, optimisation and search methods

Selection of the Master Orientation. Once the student has completed these three core subjects they will choose the focus of their masters degree. As mentioned previously, there are two possible profiles, researcher or professional skills.

Students who choose the research focus for their masters will join one of the existing research groups and choose a topic in that area. During the first part

of the course the student will research their chosen area and attend the research group meetings. This culminates in the final part of the course where they will write the masters thesis, and must defend it in front of a tribunal.

The students that follow the professional skills course will spend time working on a computational science project during an internship in a firm. They will then produce a report about the project and present it to a panel composed professors from the CAOS department.

Throughout this model students undergo a significant learning process. The three core subjects, and subsequent project give each student a well rounded knowledge of all aspects of the three Computational Science and Engineering.

2.3 Optional Subjects

The optional subjects exist to permit students from different areas to standardise their knowledge so that they may better follow the core subjects. Students with degrees not related to computational science or with bachelor degrees must choose a set of subjects in order to perform between 10 and 60 additional ECTS credits, depending on their previous knowledge. These are also the subjects that students choose from to fill the optional subject space in the course core.

There are nine optional subjects which the students may choose from, each one worth 10 ECTS credits. These subjects can be split into two groups. The first five subjects teach knowledge complimentary to the course core, and are as follows.

- **Methodology for the assignment in computational science** discusses the standards and formats used in project financing, writing computational science projects, and the presentation of results. Students learn the processes involved in applying for financing for projects and present a mock application to the other students. This is the subject where students will gain most of the knowledge they require to present their masters thesis.
- **High performance computers** describes the computer components used to create high performance machines, interconnection networks, computing clusters, fault tolerance techniques, and input/ouput mechanisms. As part of this subject students research an existing supercomputer and perform an analysis of it's components. Students also complete practical work either using a network simulator or parallel performance simulation software. Upon completing this subject the student will be able to evaluate different architectures to choose the most appropriate according to given criteria.
- **Distributed systems and GRID** introduces students to parallel and distributed systems across different administrative domains, including programming, performance optimisation, and simulation. During this subject the students will read a number of seminal papers in the field of distributed system and take turns presenting the contents to the other students, followed by discussions on these topics between the students and the instructor.
- **Application principles** teaches students to recognise different applications that need high performance computing and how to identify their requirements. In this subject the main goal is to show students the many uses of

high performance applications in industry. Students hear lectures from individuals from many different fields that use parallel tools, and can see the diverse range of areas that make use of them. Students will also choose an application and design some improvements to it, the results of which are presented to the class.

The remaining four subjects provide introductory knowledge, which enables students to follow the course core, they are listed below.

- **Operating Systems** assures that students have a standard level of knowledge of the efficient use of operating systems in scientific computation fields.
- **Scientific Computation and visualisation** teaches the basic techniques of representing computational results in different scientific fields.
- **Introduction to programming** gives students a knowledge of programming and debugging in scientific languages as well as the basic concepts of software engineering.
- **Introduction to parallel programming** establishes methodologies and strategies to paralellise applications using message passing and shared memory libraries.
- **Numerical analysis** teaches students to resolve differential equation systems and computational optimisation techniques.

Alongside the course work, there are conferences and seminars organised within the department, at which presentations are given by relevant people from a range of academic and professional areas, complementing the students' studies.

3 Experience Obtained

The Computational Science and Engineering masters programme has already begun, thus some experiences can be presented. Masters theses have been completed in different fields, including biology and forest fire propagation. There are also ongoing projects in other fields, such as social sciences and economics. These projects have been built upon areas in which the CAOS department already has experience[1]. Three examples of masters theses, which were presented and defended in July 2008, will be described here: forest fire propagation prediction[5], modelling and simulation of fish school movement[6], and parallelisation of the kriging interpolation method[7]. These topics provide good examples of fields which make use of the three pillars of computational science and engineering in order to achieve their objectives.

The prediction of forest fire propagation is possible through the use of computational science and engineering methods. In this case, science and engineering provides the moisture burn model, associated meteorological variables and other input data in order to create the propagation model. Through mathematical techniques this model can be enhanced with genetic and heuristic. Finally, computation gives the calculation power necessary to produce a set of scenarios based on these variables.

[1] http://caos.uab.es/research_projects.php

The modelling of fish school behaviour is already established and well known in the field of biological science. However, in this case the numerical model does not accurately represent reality. Using fuzzy logic it may be possible to achieve better results. Computation provides the resources required to simulate this model and the mechanisms to visualise the results in a manner that permits the biological specialists to refine their original model.

Kriging is a method for interpolating unknown values from data at known locations, while slower than other methods of interpolation it can provide more accurate results. By implementing a parallel version of kriging interpolation using message passing on a distributed memory machine it was possible to significantly reduce the execution times of the entire process.

These examples show how this course model helps specialists use computational science and engineering in order to solve their problems.

To appraise the students' degree of satisfaction the CAOS department prepared a survey where students evaluated the masters programme in general as well as each specific subject. These results are being used in the development of the second edition of this course.

4 Conclusion

This paper has presented an innovative model for teaching computational science and engineering. The focus is providing the opportunity for students with and without computational science backgrounds to be introduced to this field. Some positive results have been shown through masters theses, which demonstrate the effective application of this model.

In the second year of this course, the number of students in the programme has increased considerably, which indicates the acceptance of this model.

References

1. Guha, R., Hartman, J.: Teaching parallel processing: Where Architecture and Language Meet. In: Proceedings of Twenty-Second Annual conference on Frontiers in Education, 1992, pp. 475–479 (1992)
2. Wachter, B.: The bologna process: developments and prospects. European Journal of Education 39(3), 265–273 (2004)
3. Kargidis, T., Kefalas, P., Stamatis, D., Tsadiras, A.: Towards a European Credit Transfer System for Networked Learning (ECTS-NL) (2003)
4. European Union: ECTS Users' Guide. The European Credit Transfer System. European Commission Publication 31(03) (1998)
5. Wendt, K.: Efficient knowledge retrieval to calibrate input variables in forest fire prediction. Master's thesis, Universitat Autònoma de Barcelona (2008)
6. Gonzalez, J.: Individual oriented model applying fuzzy logic. Master's thesis, Universitat Autònoma de Barcelona (2008)
7. Pesque, L.: Solució paral.lelitzada d'interpolació kriging amb ajust automatitzat del variograma. Master's thesis, Universitat Autònoma de Barcelona (2008)

Spread-of-Disease Modeling in a Microbiology Course

George W. Shiflet and Angela B. Shiflet

Wofford College, Spartanburg, South Carolina, USA
{shifletgw,shifletab}@wofford.edu
http://www.wofford.edu/ecs/

Abstract. Microbiology is the study of microorganisms. Most college courses in microbiology emphasize the biology of bacteria and viruses, including those that are human pathogens. One challenging aspect of the course is to introduce students to epidemiology, which considers the causes, dispersal, and control of disease. Although disease transmission models have helped develop successful strategies for managing epidemics, most science students are unaware of their advantages and complexities. To address this challenge, the microbiology course at Wofford College has incorporated a sequence of three or four laboratories on modeling the spread of disease. Emphasis in Computational Science students who have studied modeling and simulation in depth serve as laboratory assistants and mentors. Evidence from test scores and self-assessment support the hypothesis that the sequence of laboratories has improved student understanding of human disease dynamics and demonstrated the utility of computational models.

Keywords: computational science, education, modeling, microbiology, spread of disease.

1 Introduction

Many institutions of higher education offer a junior/senior level microbiology course to study of the biology bacteria, fungi and viruses. Although these microorganisms are important part of various ecosystems, many can cause devastating, infectious diseases. Thus, in a course that emphasizes human disease, having epidemiology as a component of the course is essential. Epidemiology is the study of the causes, dispersal, and control of disease.

Computational models of disease transmission have been instrumental in designing successful strategies for managing epidemics for a number of diseases. For example, Marc Lipsitch in collaboration with others developed a model for the spread of Severe Acute Respiratory Syndrome (SARS) and used the model to make predictions on the impact of public health efforts to reduce disease transmission [1]. As another example, using data and mathematical models, the Dutch Ministry of Health, Welfare and Sports developed "a national plan to minimize effects of pandemic influenza" [2]. Recognizing the benefits of modeling the spread of disease, the National Institute of General Medical Sciences, one of the National Institutes of Health, has a collaborative effort, Models of Infectious Disease Agent Study (MIDAS), to develop computational

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 44–52, 2009.

models for use by "policymakers, public health workers, and other researchers who want to better understand and respond to emerging infectious diseases" [3].

To help students appreciate some of the techniques, challenges, and benefits of computational spread-of-disease models, for three years biologist George Shiflet has incorporated a sequence of laboratories into Microbiology, a class with 30 to 40 students, at Wofford College [4]. After a tutorial on using a systems dynamics software tool, students in pairs investigate various diseases, develop models of the spread of those diseases, present their work, and write an analysis of the results.

2 Tutorial in Laboratory 1

System dynamics models provide global views of major systems that change with time. Thus, such models are appropriate for studying the spread of disease. Fortunately, several easy-to-use systems dynamics tools, such as *STELLA*®, *Vensim*®, or *Berkeley Madonna*®, are available to create pictorial representations of models, establish relationships, run simulations, and generate graphs and tables of the results.

In the first week's laboratory in the sequence on modeling, students have an introduction to the fundamental ideas with a predator-prey model using the systems dynamics modeling tool *STELLA*. The concept of rate of change, or derivative, is crucial to systems dynamics modeling. Even students who have not had calculus or who have not taken mathematics in several years quickly grasp the concept and derivative notation. In the tutorial, they learn that in unconstrained growth, such as for the prey in an environment of unlimited resources and no predator, the rate of change of prey is proportional to the number of prey, or $d(prey\_population)/dt = growth\_rate * prey\_population$, where *growth_rate* is a constant. Figure 1 shows a model diagram of the prey in such a circumstance. Using systems dynamics software, the user double-clicks each component and enters the initial prey population, constant of proportionality, and differential equation for *growth*, namely *growth_rate* * *prey_population*, in the flow into *prey_population*. Then, he or she can instruct the tool to generate a table and graph, such as in Table 1 and Figure 2.

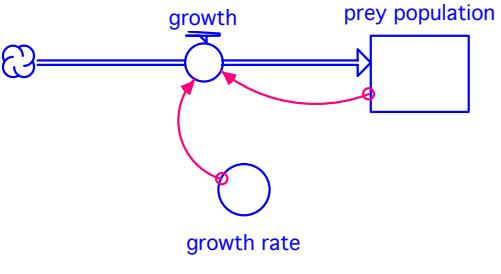


Fig. 1. Unconstrained growth diagram [5]

Table 1. Partial unconstrained growth table

Time	prey population
.00	100.00
.25	102.50
.50	105.06
.75	107.69
1.00	110.38
1.25	113.14
1.50	115.97
1.75	118.87
2.00	121.84

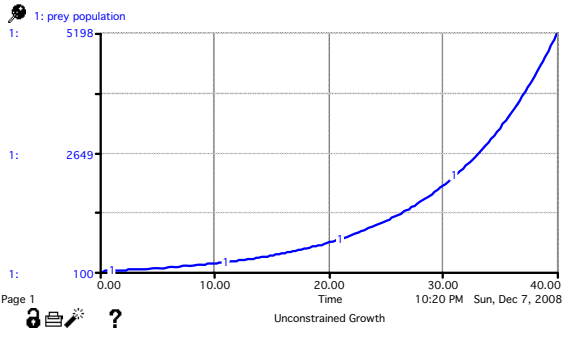


Fig. 2. Unconstrained growth graph

With the introduction of a predator, the model must consider the interaction between predators and prey. The Lotka-Volterra model has the simplifying assumptions that the particular predator only hunts the specific prey and that no other animal eats that prey. Figure 3 shows a systems dynamics diagram for this model.

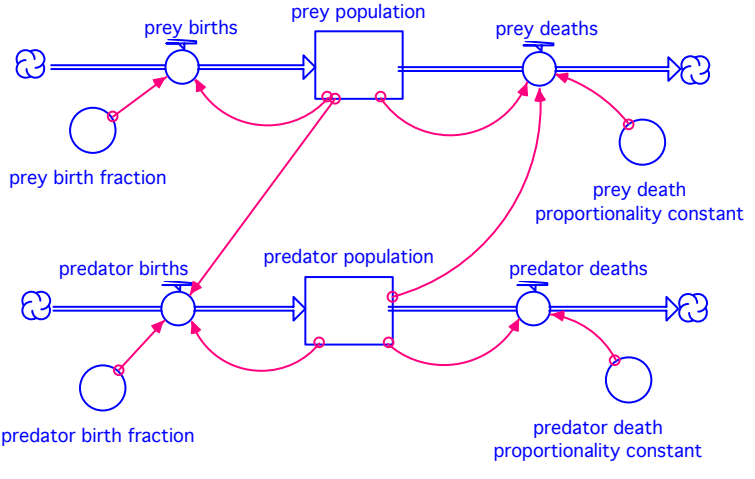


Fig. 3. Predator-prey diagram

Students quickly grasp that prey births and predator deaths follow the unconstrained growth/decay model. However, this prey's population is reduced by an amount proportional to the product of the number of predators and the prey, *prey_death_proportionality_constant * predator_population * prey_population*. One interpretation considers the product *predator_population * prey_population*, which is maximum number of distinct interactions between predators and prey. The decrease in the number of prey is proportional to this product, where the constant of proportionality is related to the hunting ability of the predators and the survival ability of the prey. A second interpretation is that the size of the prey population decreases in proportion to the size of the predator population.

While the prey population decreases with more contacts, the predator population increases. Thus, *predator_births* is *predator_birth_fraction * predator_population * prey_population*. Thus, the resulting system of differential equations for the Lotka-Volterra model is as follows, where p is the number of prey, h is the number of predators, and k_1, k_2, k_3, and k_4 are constants:

$$\frac{dp}{dt} = k_1 p - k_2 hp$$
$$\frac{dh}{dt} = k_3 ph - k_4 h$$

After considering a predator-prey model, students develop a simple SIR (susceptibles-infecteds-recovereds) model of the spread of disease using a systems dynamics tool, such as STELLA. Figure 4 displays an SIR model diagram. With the analogy of unconstrained growth/decay for prey births and predator deaths, students quickly understand the rate *recover* to be proportional to *infecteds*, so that *recovery_rate * infecteds* is the formula in the *recover* flow. Moreover, just as interactions hurt prey and help predators, interactions between *susceptibles* and *infecteds* result in illness. Thus, the flow from *susceptibles* to *infecteds* gets the formula *infection_constant * susceptibles * infecteds*. Figure 5 presents a typical graph of the number of susceptibles, infecteds, and recovereds for the SIR model.

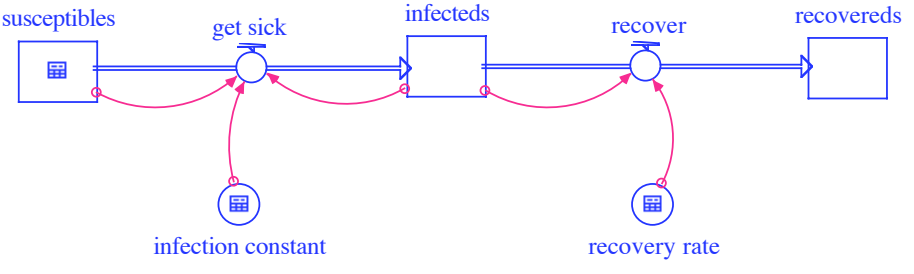


Fig. 4. SIR diagram

During this first lab, a hat is passed around the class for each student to pick a disease. With each disease listed twice, student pairs are formed to investigate their disease before the next laboratory. Between the first and second laboratories, the student pairs ascertain as much as possible about the nature of their assigned diseases, including data, such as rates of change.

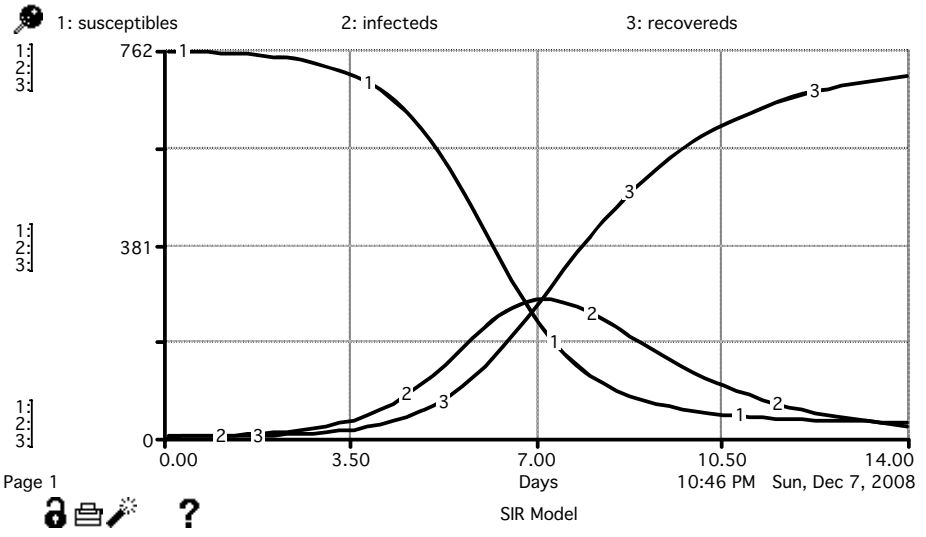


Fig. 5. Graphs for SIR model

3 Model Development in Laboratories 2 and 3

In the next two weeks' laboratories with additional time outside of class, each pair develops a model of the spread of their disease using the system dynamics modeling tool. The professor and students obtaining the Emphasis in Computational Science mentor the teams during the laboratory times and outside of class (see Figure 6) [6].

Fig. 6. Student teams

For example, one pair developed the Chagas disease model whose diagram is in Figure 7. Chagas disease is caused by the protozoa *Trypanosomiasis cruzi* (*T. cruzi*), which is usually transmitted by the feces of blood-sucking insect vectors, the "kissing bug". The disease has four main stages in a human: incubation (7-10 days), acute (3-8 weeks), indeterminate, and chronic. The only know drug must be used in the acute phase. This deadly disease infects about 14-million people worldwide, including 20% of the Bolivian population [7].

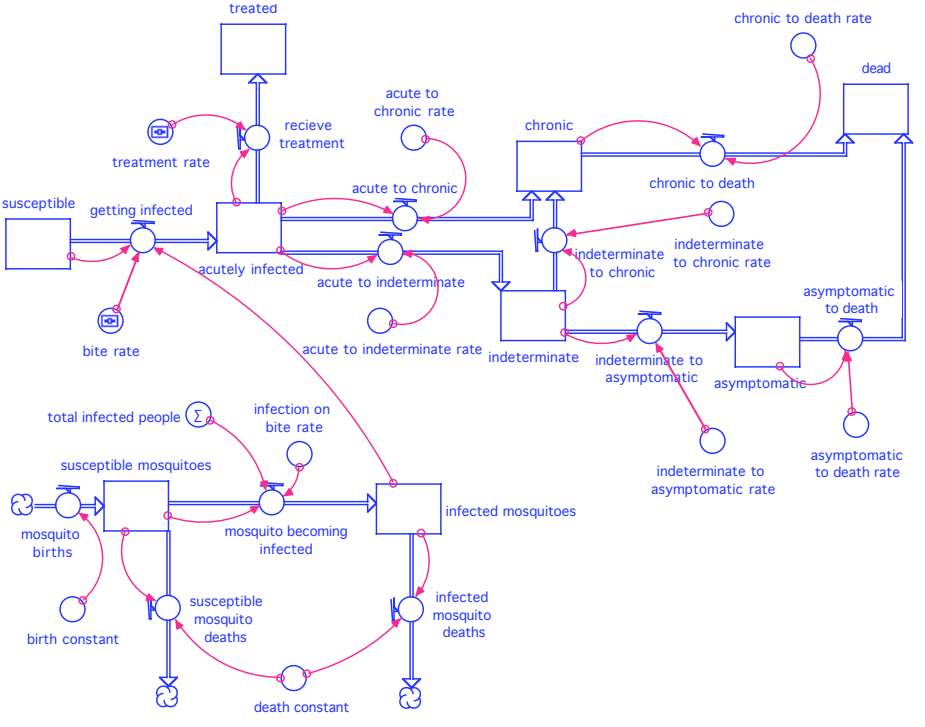


Fig. 7. Diagram model of Chagas disease by student team Lansing Yarborough and Lizzie Dilworth

4 Presentations in Laboratory 4

In the final week of the lab sequence, each team makes a 10-to-15 minute presentation on their disease and model to the class. The oral report include the following components:

- General description of the disease, including pertinent details about the agent, how it is spread, factors that affect its spread, signs, symptoms, human impact, general course, treatments, etc.
- General description of your model, including how it functions and what the team considered when designing it
- Demonstration of how the model works

- Weaknesses and strengths of the model
- Modifications that could be made to improve the model
- Applications of the model

5 Final Report

Independently, each student writes a report on his or her team's model and what they learned from the experience. These reports should include the following aspects:

- General description of the disease, which includes the basic biology of the disease (type of organism, symptoms/signs, typical treatments, etc.), a description of what is known about its transmission, and known statistics or rates
- Printout of the model along with a description of the model and how it works, the most important features and the reasons for including them, and simplifying assumptions
- Printout of execution of the model including graphs and tables of several runs of with documentation and justification for the parameters on each run
- Conclusions: What does the model show? What factors does it take into account? What factors seem to be most important to the spread of this disease? What factors does it not take into account and why? How could the model be improved? What applications might be made of the model?
- References for learning about the disease and developing the model along with a briefly critique for utility and quality of each reference
- Self-assessment: How does the individual think he or she performed as a modeler? Did developing the model help the student to understand about the disease and its transmission? How? If not, why was it not helpful?

One or two computational science students help in evaluation of the models and the reports.

6 Evaluations

The assessment portions of the final reports reveal a deeper understanding of the spread of their diseases, the modeling process, and the utility of models. The following comments are representative:

- "If I had not already chosen a profession such a long time ago, this [computational biology] would certainly be a possibility. I feel that by designing a model I gained a more comprehensive understanding of Chagas Disease than I would have if I simply had to do a PowerPoint presentation on the subject. The model almost allowed an 'inside-look' at the mechanics of the disease."
- "Developing the model helped me to appreciate the inter-connectedness of all of the factors that influence a disease and helped me to visualize why the disease becomes endemic. I enjoyed the victory of understanding something that initially made little sense to me and frustrated me."

- "The model definitely helped me to understand more of pneumocystis and its transmission. Only so much of a disease can be understood from a textbook, especially of its complexities. Software as dynamic as Stella makes learning about the way pneumocystis is transmitted so much clearer and more comprehensible. I wasn't only seeing numbers as I would on a page, but literally watching the rates and progress of transmission change before my eyes."
- "Developing this model was very useful in learning about the spread of Lassa Fever. Through research, I was able to learn about this disease while at the same time applying what I learned to the model. Having to make a model based on what you learn makes you very conscious of all the factors that influence the spread of the disease."
- "I did enjoy constructing the model and would definitely enjoy taking on the challenge of developing a model for a much more complicated disease. Developing the model did help me understand the disease and its transmission….I feel that this exercise has been a beneficial experience for me and is an excellent tool for studying disease and epidemiology."

Although evaluations by the students are almost unanimously favorable, improved performance by the students is difficult to quantify. However, on the microbiology final exam, the professor asks questions about disease risks and preventative measures when traveling abroad, and the answers now include in-depth considerations of vector control, water purification, and the other less obvious complicating factors. Before starting the modeling component of the course, students just gave a list of diseases with avoidance behavior suggestions. Moreover, in a subsequent medical case studies class, students present much more comprehensive suggestions to management and prevention issues than previously provided. Thus, the sequence of modeling laboratories appears to accomplish the goals of improving students' understanding of human disease dynamics and the utility of computational models.

7 Conclusion

Modeling the spread of disease in a sequence of microbiology laboratories has been beneficial in a number of ways. Students gain an understanding of fundamental concepts, such as rate of change, unconstrained growth, and interactions. Through model development, testing, and refinement, they utilize and improve critical thinking and problem-solving skills. By working with a partner, the students experience teamwork, which is so important to science. Through project oral presentations and written reports, they can improve their communication skills. Moreover, we have found that this interactive learning experience enhances the students' appreciation and understanding of modeling and computational science.

References

1. Lipsitch, M., et al.: Transmission Dynamics and Control of Sever Acute Respiratory Syndrome. Sciencexpress Report, May 23 (2003), http://www.sciencexpress.org/23May2003/Page1/10.1126/science.1086616
2. van Genugten, M.L.L., Heijnen, M.A., Jager, J.C.: Pandemic Influenza and Healthcare Demand in the Netherlands: Scenario Analysis. Emerging Infectious Diseases 9(5), 531–538 (2003)

3. Models of Infectious Disease Agent Study - National Institute of General Medical Sciences, http://www.nigms.nih.gov/Initiatives/MIDAS/
4. Wofford College, http://www.wofford.edu
5. Shiflet, A., Shiflet, G.: Introduction to Computational Science: Modeling and Simulation for the Sciences. Princeton University Press, Princeton (2006)
6. Computational Science, http://www.wofford.edu/ecs/
7. Chagas Disease (American Trypanosomiasis): Overview - eMedicine, http://emedicine.medscape.com/article/214581-overview

An Intelligent Tutoring System for Interactive Learning of Data Structures*

Rafael del Vado Vírseda, Pablo Fernández, Salvador Muñoz, and Antonio Murillo

Departamento de Sistemas Informáticos y Computación
Universidad Complutense de Madrid, Spain
rdelvado@sip.ucm.es,
{pablo.fdez.p,salva.ms,murillo925}@gmail.com

Abstract. The high level of abstraction necessary to teach *data structures* and *algorithmic schemes* has been more than a hindrance to students. In order to make a proper approach to this issue, we have developed and implemented during the last years, at the Computer Science Department of the Complutense University of Madrid, an innovative *intelligent tutoring system* for the interactive learning of data structures according to the new guidelines of the *European Higher Education Area*. In this paper, we present the main contributions to the design of this intelligent tutoring system. In the first place, we describe the tool called *Vedya* for the visualization of data structures and algorithmic schemes. In the second place, the *Maude* system to execute the algebraic specifications of abstract data types using the *Eclipse* system, by which it is possible to study from the more abstract level of a software specification up to its specific implementation in *Java*, thereby allowing the students a self-learning process. Finally, we describe the *Vedya Professor* module, designed to allow teachers to monitor the whole educational process of the students.

1 Introduction

The study of *data structures* and *algorithmic schemes* constitutes one of the essential aspects of the academic formation of every student in Computational Science. Nevertheless, the high level of abstraction necessary to teach these topics occasionally hinders its understanding to students. In order to make a proper approach to this issue, we have developed and implemented during the last years, at the Computer Science Department of the Complutense University of Madrid, an innovative interactive and visual learning framework according to the new guidelines of the *European Higher Education Area* and the teaching model focused on the student.

Our innovative approach is based on an *Intelligent Tutoring System* [8] (shortly, ITS), a computer system that provides direct customized instruction

* This work has been partially supported by the Spanish National Projects FAST-STAMP (TIN2008-06622-C03-01), MERIT-FORMS (TIN2005-09027-C03-03) and PROMESAS-CAM (S-0505/TIC/0407).

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 53–62, 2009.

and feedback to our students, a personal training assistant, and a range of tutoring techniques according to the student's response without the intervention of human beings. Thus, our ITS implements the underlying theory of *Abstract Data Types* [5] by doing and enable students to practice their skills by carrying out tasks within highly interactive learning environments. Based on these learner tools, the ITS tailors instructional strategies, providing explanations, hints, examples, demonstrations, and practice problems on data structures and algorithmic schemes [1,4,7]. The evaluation of our research on that systems indicates that students taught by our ITS generally learn faster and translate the learning into improved performance better that classroom-trained participants.

Despite the concept of ITS has been pursued for more that thirty years by researches in education, psychology, and artificial intelligence, few systems are in practical use today for the interactive learning in Computational Science. In order to remedy this lack, this paper describes the design of an ITS which guides the interactive learning of data structures from the *algebraic specification* to the real implementation [5]. The main components of this system are, on the one hand, the *Vedya tool* [9], a visualization tool by means of which, it is possible to provide the students with a complete learning system of both, the main data structures and the more relevant algorithmic schemes. On the other hand, the *Maude system* [3] for the execution of algebraic specifications of abstract data types using the language of formal specification provided by this system. And third, thanks to the development environment of *Eclipse*, we have obtained a fully complete system that is useful for the students as well as the professors, that allows to go from the most abstract level of data structures, provided by its algebraic specification in *Maude*, until its specific implementation in a modern programming language as happens with *Java*. All this learning process can be guided and overseen in a completely autonomous way by using the ITS presented in this paper, through which it is possible to make enquiries about the documentation related to each of the algebraic specifications, to distinguish between the behavior of the data structure and its different implementations through the use of different views or to browse information regarding the cost of the different implementations that have been proposed.

2 The Vedya Tool

Vedya is an integrated interactive environment for learning data structures and algorithmic schemes presented for the first time in [9]. It covers the most common data structures: Stacks, queues, binary search trees, AVL trees, priority queues, and sorted and hash tables. Moreover, it also provides other different types of abstract data types, like one for an implementation of a "doctor's office". Concerning the algorithmic schemes, it covers the most common resolution methods [1,4,7]: Divide and conquer, dynamic programming, backtracking, and branch and bound. All data structures and algorithmic schemes taught in the related study courses are thereby integrated in the same environment.

Currently, there are two versions of the *Vedya* tool. The first version contains all the data structures and algorithmic schemes mentioned above while the new

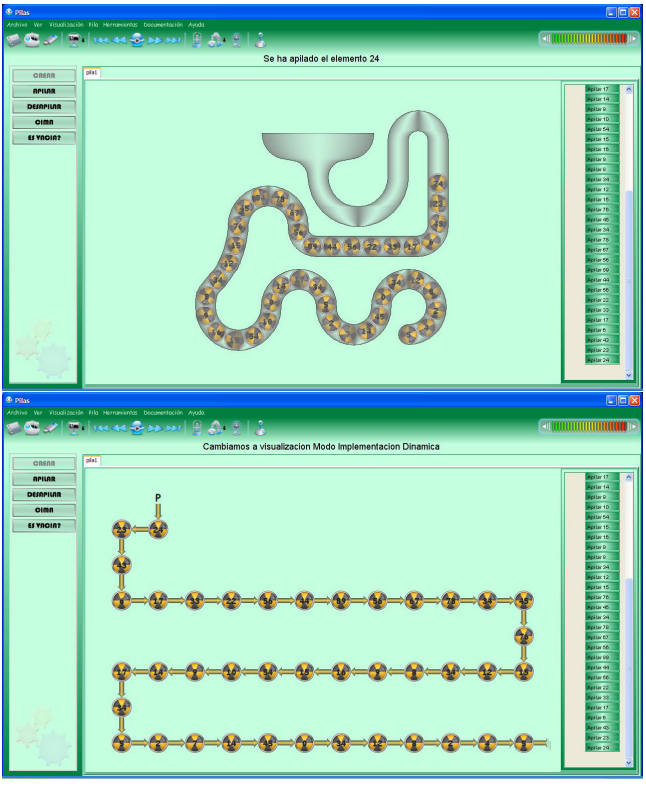


Fig. 1. Data structures in the Vedya tool

one offers a subset of them in a more attractive visual environment. This last version can be found at `http://www.fdi.ucm.es/profesor/rdelvado/Vedya.zip`.

There are several options to use this tool. The main one is the interactive execution, but it is also possible to create simulations that are automatically executed, to visualize tutorials and to solve tests within the same environment. It also integrates a set of animations that show how data structures are used to solve certain problems. For instance, Fig. 1 shows an example of the main windows for stacks. The central panel is used to represent the structure. On the left, there is a list of the actions that can be executed. Partial non-allowed actions are disabled. The right panel shows the visualization of the actions that have been already executed. There are two types of views: The one of data structure behavior to intuitively comprehend its operation, and one or several implementation views, either static or dynamic. On Fig. 1 we show the specific behavior view of a stack and a dynamic implementation based on pointers. The environment also provides documentation about algebraic specification, the implementation code and the cost of each implementation. Moreover, the current version of *Vedya* offers the *Vedya-Test* tool to solve tests (see Fig. 2). This tool can be independently executed and

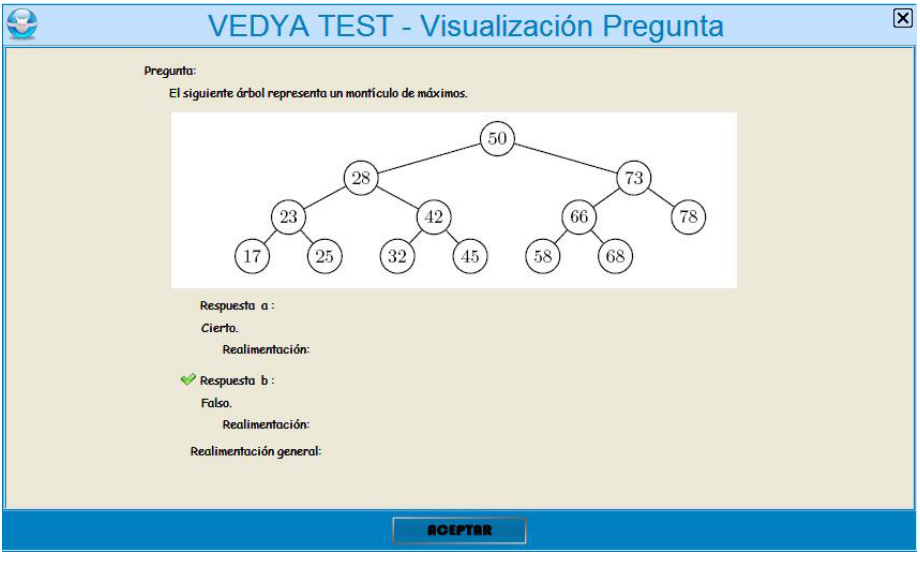


Fig. 2. The Vedya-Test tool for the student evaluation

allows teachers to create, modify or delete questions in a database. The student visualizes the tests, solves them and obtains the correct solutions. Questions are grouped by subject-matter on the database, but it is possible to mix questions about different data structures in the same test. The last version of this tool can be also found at `http://www.fdi.ucm.es/profesor/rdelvado/vedya-test.zip`.

3 Execution of Algebraic Specifications in Maude

For the execution of algebraic specifications in our ITS, we use the language *Maude* [3] based on *rewriting logic*. *Maude* is a high-level language and high-performance system supporting both equational and rewriting computation for a wide range of applications. *Maude* and its formal tool environment can be used in three mutually reinforcing ways: as a declarative programming language, as an executable formal specification language, and as a formal verification system. Moreover, [6] describes the equational specification of the data structures included in the *Vedya* tool now in *Maude* syntax (stacks, queues, lists, binary and search trees, AVL and 2-3-4 trees). The language is available for *Linux* and *Mac-OS* at `http://maude.cs.uiuc.edu`, and there are also extensions for its execution in *Windows* at `http://moment.dsic.upv.es`.

The algebraic specifications can be efficiently executed in the *Eclipse* system (`http://www.eclipse.org/`) by means of special "plugins" (which can be downloaded from `http://www.fdi.ucm.es/profesor/rdelvado/plugins-eclipse.zip`) developed in the Department of Information Systems and Computation of the Technical University of Valencia (DISC-UPV) and in the Computational Languages and Sciences Department of the University of Málaga (DLCC-UMA).

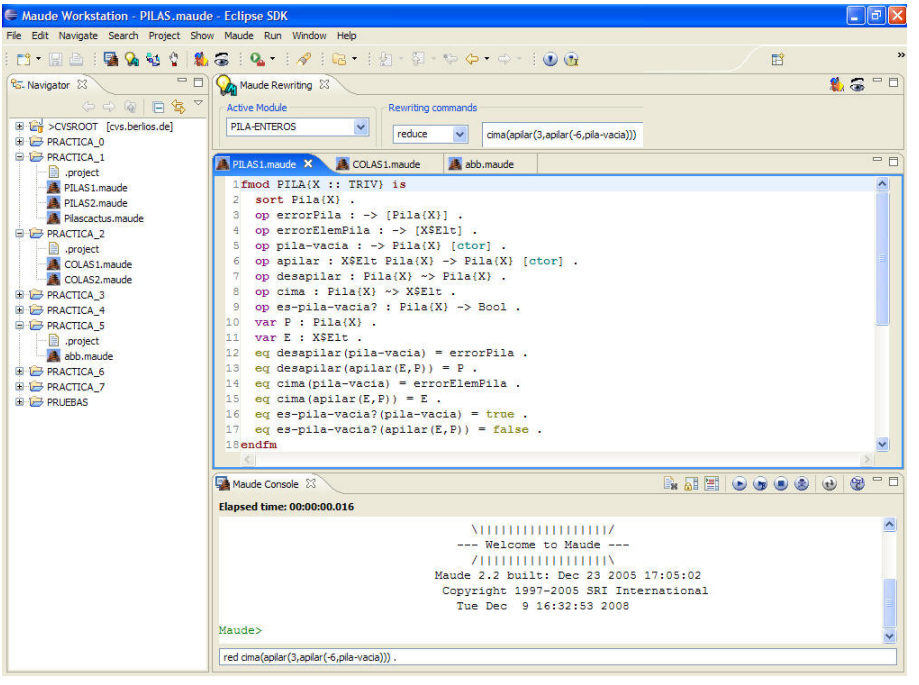


Fig. 3. Integration of Maude in Eclipse for the execution of algebraic specifications

This environment, as shown in Fig. 3, facilitates the student its usage by integrating the text editor with the execution commands of the system. On the left, there appear the developed projects; the central part shows the editor and the execution panel of the system is on it; on the inferior part, the control panel that shows the result of the action.

The basic element of a specification in *Maude* is a "module". The language allows defining the functional modules used to define data types. For example, the functional module for stacks used in Fig. 3 is showed with more detail in Fig. 4. The modules can be customized, using "theories" to such end in order to define the parameters and "views" to relate the formal parameter to the real parameter. The system has predefined the abstract data types most commonly used, as well as the most common theories and views:

```
view Int from TRIV to INT is
  sort Elt to Int .
endv
```

```
fmod STACK-INTEGERS is
  including STACK{vInt} .
endfm
```

In order to execute the specification, the student enters the text given in Fig. 4 in the editor of *Eclipse* (see Fig. 3); then, she/he executes the *Maude* system using

```
fmod STACK{X :: TRIV} is
     sort Stack{X} .
     op   error       :                -> Stack{X} .
     op   error       :                -> X$Elt .
     op   empty       :                -> Stack{X} .
     op   push        : X$Elt Stack{X} -> Stack{X} .
     op   pop         : Stack{X}       -> Stack{X} .
     op   top         : Stack{X}       -> X$Elt .
     op   isEmpty?    : Stack{X}       -> Bool .
     var  P           : Stack{X} .
     var  E           : X$Elt .
     eq   pop(empty)          = error .
     eq   pop(push(E,P))      = P .
     eq   top(empty)          = error .
     eq   top(push(E,P))      = E .
     eq   isEmpty?(empty)     = true .
     eq   isEmpty?(push(E,P)) = false .
endfm
```

Fig. 4. Algebraic specification of parametric stacks in Maude syntax

the existing buttons in the *Maude Console* of *Eclipse* and enters the module. The system detects existing syntax errors and shows them on the *Maude Console*. Once the module shows no more errors, the student may reduce terms by using the equations of the module. To such end, the student may use the commands chart placed at the top of the screen or she/he may directly write the command in the editor and enter it into the system. For example, in order to obtain the top of a stack, we can reduce the term: `red top(push(push(empty,5),4)).` This term must be reduced over the module of the stacks using the integer number theory `INT`. In our example, this module is named: `STACK-INTEGERS`.

The possibility of reducing terms, in an automatic way, allows the students to carry out an initial test of their specifications by detecting many of the errors committed when defining the operations using equations. Another greater advantage of executing the specifications is that the student comprehends the difference between the parameterized module and the instantiated module by being able to reduce terms on different modules. For example, a new module could be named `STACK-CHARACTERS` on which terms of type `red top(push(push(empty, 'a'),'c')).` can be easily reduced. Other examples of abstract data types, such as a "doctor's office" can be proposed [5]. In all of them, the aim was to define parameterized or instantiated data type with different theories. The practical classes are complemented with different terms that the student must reduce over some type of instantiated modules to prove the specification, as well as proposals to make little changes in some actions or erroneous definitions to detect them (see `http://www.fdi.ucm.es/profesor/rdelvado/codigo-maude.zip`).

Taking into consideration that students from the second year were involved, just a few of the language facilities have been used. In superior courses where students have more knowledge on the subject, a richer language can be used [3]

(e.g., many-sorted equational specifications, order-sorted equational specifications, equational attributes, and membership equational logic specifications).

4 An Intelligent Tutoring System for Data Structures

An *Intelligent Tutoring System* (shortly, ITS) for the *Vedya* tool turns into a pedagogical instrument of high practical interest since it attempts to address the whole self-learning process of the main data structures, from the algebraic specification written in *Maude* until the real implementation written in *Java*, within such a powerful and integrated environment as the *Eclipse* system described in the previous section.

The students have their first contact with the data structures that they are going to study by means of the usage of the ITS on *Vedya*. In order to control the student's self-learning process correctly, an online database has been built in on this tool. This means that now, the user has to be logged before using the *Vedya* tool, in order to oversee he/she evolution properly. For this purpose, the additional module called *Vedya Professor* (which can be obtained from `http://gpd.sip.ucm.es/rafav/`), has been designed to take full advantage of this feature. This tool allows teachers to monitoring the current progress of their students as a whole (see Fig. 5), according to the information stored in the database (tests realized, time spent on each test or most consulted documents

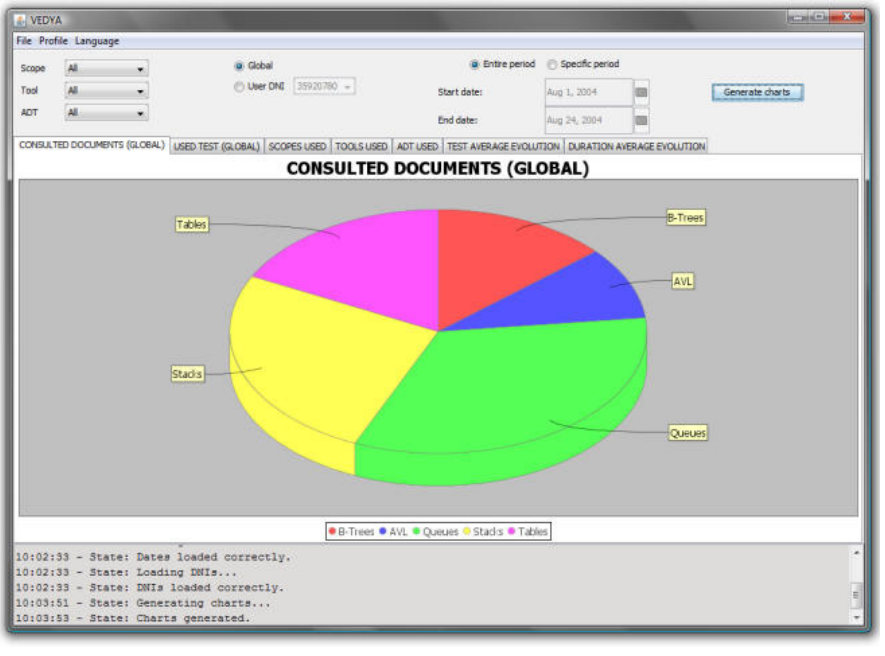


Fig. 5. Vedya Professor Module for the Vedya tool

on the help of *Vedya*). Moreover, this tool also allows seeing detailed information of each specific student, by selecting their identification number.

For example, if their learning of data structures is focused on linear data structures or binary search trees, the ITS would suggest that the student should start their learning process in the corresponding section of the tool, where they will be able to experiment, freely and on their own, each one of the actions offered by these structures (see Fig. 1). In order to strengthen and evaluate this intuitive knowledge, the student has, in addition, the useful possibility of using the *Vedya-Test* tool (see Fig. 2).

Once the student has a clear idea of the informal behavior of the data structure, the ITS may continue working on the *Eclipse* system. The first step would be to formally capture the intuitive knowledge that the student has obtained through the usage of *Vedya* in a specific algebraic specification written in *Maude* syntax. In order to facilitate this difficult step in the student's self-learning, the student may use, interactively, the documentation that is included in the manual of the *Vedya* tool.

Once the algebraic specification in *Maude* syntax (see Fig. 4) is entered into the *Eclipse* system (see Fig. 3), the student can now go on executing little tests using the *Maude Console*, in order to check whether it coincides with the intuitive and informal notion of data structure from which he/she initially departed in *Vedya*. Such experience would allow the student to reach the high level of abstraction that is necessary in computer supported education for each formal specification of a software component, always based on the intuitive and experimental knowledge.

Once the algebraic specification of the data structure is obtained, the next step performed by the ITS would be to develop an implementation in an object-oriented programming language such as *Java*, by means of the facilities provided by the programming environment in *Eclipse*. This time, the student may use the algebraic specification that she/he has built, as if dealing with an authentic "instructions manual". The main advantage of our methodology is that the specification behaves now as a prototype of the data structures to be implemented, in a way that the student is able to find out the exact behavior for all those moments of doubt that may appear during the design process, even before the student is able to compile their programs. In order to be able to guide, in a more specific way, the step from specification to implementation, the student may make use again of the *Vedya* tool. This time, the student may access to the part that would correspond with the implementation of the data structure that she/he is studying from the options menu (see Fig. 1). From there, he/she may try different implementation possibilities based on arrays or pointers.

Once the student is familiar with the different implementations of the structure, she/he is finally ready to properly decide on a suitable representation in the *Java* language. The possibility of having understood and previously evaluated the different implementations by means of the ITS allows the student the possibility to acquire a clear knowledge of the *algorithmic cost* of the chosen implementation in *Java* for each specific operation of the data structure, so that

this would also be a decisive criterion at the moment of designing its own implementations. In this part, the "*algorithmic schemes*" part of the *Vedya* tool plays an important role, since it allows the student to acquire a good programming methodology.

5 Evaluation

In order to obtain a detailed evaluation of the usage of the ITS on *Vedya* and *Maude* in our integrated *Eclipse* system, we have proposed several tests (see `http://www.fdi.ucm.es/profesor/rdelvado/Tests.zip`) related to the behavior, specification, implementation and application of the main data structures offered by the tool in the "Data Structures" academic subject at the second year, and in the "Programming Methodology and Technology" subject at the third year.

Taking into account this profile of our engineering and Computer Science students, we have proposed 8 tests in the *Virtual Campus* of the Complutense University of Madrid (`http://www.ucm.es/campusvirtual/CVUCM/`). The number of engineering students registered in this *Virtual Campus* was just over 122. Fig. 6 shows the number of the students who answered each of the tests. We observe that, from the third test on, the number of students becomes stable in a number lightly low to the number of students who access regularly to the *Virtual Campus*. These numbers, though seemingly high, are only between 30% and 40% of registered students, which shows the high rate of students giving up in this topic from the beginning. Fig. 6 also shows the percentage of correct answers: In general, it is high, which demonstrates the interest of the students who have taken part. Fig. 7 shows the percentage of students that did not attend the final exam, those who passed, and those who failed during the last six years. We observe that in the last academic courses, in which we have applied the ITS on *Vedya* tool, we have reduced by 10% the percentage of students giving up the course with respect to the previous course, and at the same time, we have increased by 12% the percentage of students that passed the exam. The percentage of students that failed the exam decreased by 2%. Comparing with previous

	Stacks 1	**Stacks 2**	**Queues**	**Sequences**	**BST**	**AVL**	**RB**	**Heaps**
Students	65	61	57	31	35	38	32	39
Answers	76.4%	82.5%	77.8%	65.6%	82.2%	84.9%	80.2%	86.3%

Fig. 6. Students answering the tests and percentage of correct answers

	2002/03	**2003/04**	**2004/05**	**2005/06**	**2006/07**	**2007/08**
Not attended	57.6%	45.3%	42.3%	64.7%	50.8%	40.2%
Passed	15.3%	22.2%	20.2%	18.2%	30.1%	42.6%
Failed	27.1%	32.5%	37.5%	17.1%	18.9%	17.2%

Fig. 7. Comparison of academic results with previous courses

courses (2003 to 2004) the percentage of students that passed has increased between 20% (with respect to the course 2003/04) and 25% (with respect to the course 2002/03).

6 Conclusions and Future Work

In the last years, many papers on visualization of data structures have been written (see, e.g., [2]). Nevertheless, there is a lack in many of them of a tutoring system which guides the interactive learning of data structures from the algebraic specification to the real implementation by means of appropriate user tools.

In this paper, we have described the design and usage of an innovative educational environment for the interactive learning of data structure by means of an *Intelligent Tutoring System* [8]. This system can be efficiently applied on the visualization tool *Vedya* [9] and the specification language *Maude* [3] with its programming environment in the *Eclipse* system, allowing the students the possibility of acquiring the capacity of implementing, correctly and properly, a data structure according to its formal algebraic specification, using in their design, the proper algorithmic schemes. As a consequence, it is possible to provide the students with a complete and professional methodology of software development that is very useful in the current teaching of Computer Science.

As future work, we plan to integrate, as part of the development of our intelligent tutoring system, an interface of the current "*Vedya-Maude*" system in *Eclipse*, in order to control and guide students along their self-learning process in a more autonomous way.

References

1. Brassard, G., Bratley, P.: Fundamentals of algorithms. Prentice Hall, Englewood Cliffs (1996)
2. Chen, T., Sobh, T.: A tool for data structure visualization and user-defined algorithm animation. In: Frontiers in Education Conference (2001)
3. Clavel, M.: All About Maude - A High-Performance Logical Framework. LNCS, vol. 4350. Springer, Heidelberg (2007)
4. Cormen, T., Leiserson, C., Rivest, R.: Introduction to Algorithms. The MIT Press, Cambridge (2001)
5. Horowitz, E., Sahni, S., Mehta., D.: Fundamentals of Data Structures in *C++*. W.H. Freeman & Co., New York (1995)
6. Martí, N., Palomino, M., Verdejo, A.: A tutorial on specifying data structures in Maude. Elsevier ENTCS 137(1), 105–132 (2005)
7. Neapolitan, R., Naimpour, K.: Foundations of algorithms using C++ pseudocode. Jones and Bartlett (2003)
8. Psotka, J., Mutter, S.A.: Intelligent Tutoring Systems: Lessons Learned. Lawrence Erlbaum Associates, Mahwah (1988)
9. Segura, C., Pita, I., del Vado, R., Saiz, A., Soler, P.: Interactive Learning of Data Structures and Algorithmic Schemes. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part I. LNCS, vol. 5101, pp. 800–809. Springer, Heidelberg (2008)

A Tool for Automatic Code Generation from Schemas

Antonio Gavilanes, Pedro J. Martín, and Roberto Torres

Departamento de Sistemas Informáticos y Computación, Facultad de Informática,
Universidad Complutense de Madrid, 28040 Madrid, Spain
{agav,pjmartin}@sip.ucm.es, r.torres@fdi.ucm.es

Abstract. Algorithm design is one of the more neglected aspects in programming introduction courses. On the contrary, schemas focus on solution construction, since they gather common characteristics of algorithms, so they can be considered as algorithm cognitive units. In this paper, we go beyond the benefits of teaching schemas and we present a tool that incorporates their use. It automatically generates code from the application of schemas, allowing its integration into the class as a useful educational tool.

Keywords: Automatic code generation, schemas, recurrence relations.

1 Introduction

A first year programming course usually is devoted to instructing students on two different aspects: the programming language and the algorithm design. Most of the textbooks focus on the language itself, thus algorithms are scattered along the course to show the language features. As a consequence, the algorithmic knowledge is poorly organized and students find that each problem requires an innovative technique to be solved.

On the contrary, not so frequent trends structure students' instruction around problem analysis and solution construction, by means of teaching schemas [6] [8] [9]. Schemas join the common characteristics of the algorithms that solve a family of problems, thus they can be considered as algorithmic cognitive units that can be applied to build programs. Students must carefully analyze the problem to find the schemas that can be applied to solve it, instead of programming from scratch. This analysis is based on drawing analogies to identify the tasks whose solutions are well known [5]. For instance, when analyzing the query "*is x prime?*" or the calculation of "*trunc($log_2(x)$)*", for a given natural number $x \geq 2$, students should notice that both problems can be similarly solved by using a search schema. Indeed, we can look for the first natural number $y \geq 2$ such that *y divides x*, for the first one, and *pow(2,y)>x*, for the second one, where *pow(x,y)* computes x^y. In fact, students should be able to instance a skeleton like the following one, using Java syntax:

```
y= 2;
found= CONDITION;
while (!found) {
  y= y+1;
  found= CONDITION;
}
```

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 63–73 2009.

where `CONDITION` respectively corresponds to the expressions `x % y == 0`, and `pow(2, y)>x`. After execution, we use "`x is prime iff y=x`" to solve the first problem, and return `y-1` for the second one.

Teaching schemas has great benefits. Regarding students, it improves their ability for abstraction, it avoids a compulsive impulse to write code before knowing what to do; and it standardizes the code that different students could produce. From a teaching point of view, instructors can exploit a broad analysis of the schemas in order to automatically extend their properties to any solution based on them. Hence properties such as correctness, termination or complexity, can be stated once in a theoretical framework, instead of independently analyzing each program. Also, schemas provide students with important insights into the use of other algorithmic units, such as design patterns in later courses.

In this paper we go beyond teaching schemas, since we also take care of how they can be automatically applied to a given problem. Apart from teaching how to instantiate the variable parts of the schema (e.g. `CONDITION` in the examples above), we use a tool to generate the involved code. Thus, when students are asked to solve a problem, firstly they should represent the problem in order to supply it to the tool, and then, choose the proper schema. So the tool allows the student to focus on the schema, not on the syntax of the language, and autonomously to obtain running solutions to the problem, from the code that is automatically generated. Visual and iconic languages, and their programming environments, are also related to code synthesis for programming instruction [2] [3] [13]. But they are based on graphical description of the algorithms, thus our approach has a greater abstraction power, since it requires the specification of the problem instead.

2 Theoretical Framework

In an introductory programming course, schemas can be mainly used to solve two different tasks: *traversing* and *searching*. As we have seen, the primality test is an example of a searching process. The computation of *pow(x,y)* itself can be seen as an example of a traversal from *1* to the natural number *y*. Nevertheless, schemas must also be classified according to the way data are generated. We have been teaching them in three contexts: data built by recurrence relations, data obtained from an array, and data read from a file. Since the schemas involved in the exploration of recurrence equations are the simplest ones, we began developing a tool to solve them.

2.1 Recurrence Relations

In mathematics, a *recurrence relation* is an equation which defines a sequence recursively: each term of the sequence is defined as a function of the preceding terms [4]. To obtain a unique sequence from a recurrence relation, there must be some initial values that do not depend on other numbers in the sequence. A well-known example of recurrence relation is the Fibonacci sequence given by the equation $f_i = f_{i-1}+f_{i-2}$ and the initial values $f_0=1$, $f_1=1$. The *order* of a recurrence relation is the number of preceding terms occurring in the equation; so the order of the Fibonacci sequence is 2. The *index* of a term is its position in the sequence, beginning from *0*.

We do not intend to solve such relations, as usual in a discrete mathematics course. Actually, we are concerned about generating iterative algorithms to explore the

recurrence sequence. Thus, we do not care about the type of its terms –`float`, `int` or `boolean`-, nor the operators involved in the equation. We will only suppose that the related expressions are valid. As operands of the equation defining f_i, we allow not only the preceding terms of f, but also the index i itself, and the preceding terms of other recurrence relations. In the latter case, it is said that the relations have been simultaneously defined by a *recurrence relation system*. For instance, the system $f_0=1$, $g_0=0$, $f_i=g_{i-1}$, $g_i=f_{i-1}$, defines the characteristic functions of the predicates "*i is even*" (f) and "*i is odd*" (g). The order of a system is the maximum of the orders of its relations. Nevertheless, we will only consider systems whose relations have the same order. Systems not satisfying this condition can be completed by progressing on the relations fallen behind the rest. Finally, we also allow systems where f_i depends on the i-th term of a simultaneous relation. In order to avoid partiality in this case, a topological ordering between the relations of the system is required. For example, equations $f_0=0$, $g_0=0$, $f_i=g_i+f_{i-1}$, $g_i=f_{i-1}+g_{i-1}$ compose a proper system since g_i can be computed before than f_i.

Apart from the types and operators involved in the recurrence relations above presented, the computability they define can be compared to the class of primitive recursive functions [7].

2.2 The Schemas

We use schemas to solve three classic problems involved in the exploration of recurrence relation systems: (1) the *traversal problem*, that calculates the term occurring at a given index, (2) the *unbounded search problem*, which looks for the first term satisfying certain condition, and (3) the *bounded search problem*, which seeks the first term satisfying a condition up to a given index. Among the different schemas that can be designed to solve such problems, we present the following ones using Java syntax.

```
//TRAVERSAL
int i;
DECLARATION
INITIALIZATION
i= CURRENT_INDEX;
while (i<n) {
  i= i+1;
  STEP
}
```

```
//UNBOUNDED SEARCH
int i;
boolean found;
DECLARATION
INITIALIZATION
i= CURRENT_INDEX;
found= CONDITION;
while (!found) {
  i= i+1;
  STEP
  found= CONDITION;
}
```

```
//BOUNDED SEARCH
int i;
boolean found;
DECLARATION
INITIALIZATION
i= CURRENT_INDEX;
found= CONDITION;
while (!found &&(i<bound)){
  i= i+1;
  STEP
  found= CONDITION;
}
```

The three schemas use a `while` sentence to progress on the recurrence system, step by step. The variables `i` and `found` are related to the last computed terms of the system and denote their index and whether they satisfy `CONDITION`, respectively. Capitalized words are "holes" that must be properly replaced depending on the given recurrence system. `DECLARATION` and `INITIALIZATION` must be replaced with the corresponding variables, `CURRENT_INDEX` must be replaced with the order of the system minus one, and `STEP` must be replaced with the code required to progress on the system.

3 The CGR Tool

The CGR tool, which stands for *Code Generation for recurrence Relations*, produces Pascal, C and Java code (the *target* languages) from a specification of the involved recurrence relations. We present how it works by means of examples, and we show how its GUI looks by displaying different snapshots.

3.1 Example 1. Computing the Vertices of a Regular N-gon

As a first example, consider the problem of calculating the vertices of a regular *N*-gon (*N>2*) for a given side *length>0*. In order to apply the tool, the student must begin defining the recurrence relations required to solve the problem, which basically correspond to express how vertices coordinates develop. The solution we propose is based on the well-known turtle graphics [1]: assuming that the first vertex is placed at an initial arbitrary point, the coordinates of the next vertex yield after properly rotating the direction and moving forward the distance *length*. Let a_i be the recurrence defining the angle that must be rotated. It can be defined by:

$$a_0=0$$
$$a_i=a_{i-1}+2\pi/N, \quad i>0$$

The recurrences x_i and y_i define the coordinates of the successive vertices. If we start at the point `(0, 0)`, they can be defined as follows:

$$x_0=0,\ y_0=0$$
$$x_i=x_{i-1}+length*\cos(a_{i-1}), \quad i>0$$
$$y_i=y_{i-1}+length*\sin(a_{i-1}), \quad i>0$$

Each recurrence relation is provided to the tool by using a dialog box which requests the following inputs from the students: the name of the recurrence, the primitive type (`real`, `integer`, `boolean`) it holds, the expressions for the initial values, and the equation for the recurrence relation (Fig. 1).

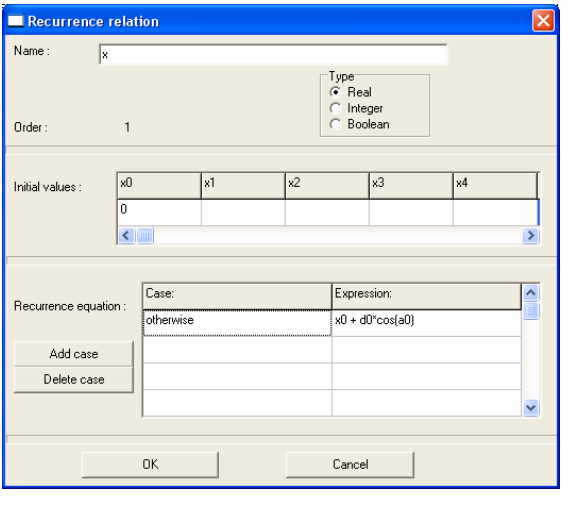


Fig. 1. Defining the recurrence relation `x`

In the generated code, identifiers `f0,…,fm-1` will hold the last generated terms of a given recurrence relation `f` of order `m`; so they must be used to state its recurrence equation. In the example, the identifier `x0` denotes the last term of `x`, and `x0` must be used to state the involved recurrence equation: `x0+length*a0` (Fig. 1). We tell students that they must instance the equation to compute the first new term ($x_1=x_0+length*\cos(a_0)$ in the example), when they provide CGR with the recurrence equations. Once students have defined the recurrences that compose the system, they have to determine which schema must be applied to solve the problem, and which target language (Pascal, C, Java) will be used in the code generation. This information is supplied to the tool by using a new dialog box which requires the names of the recurrences and the chosen schema(s). In our example, we must apply the traversal schema since we ask for all of the *N*-gon vertices (Fig. 2).

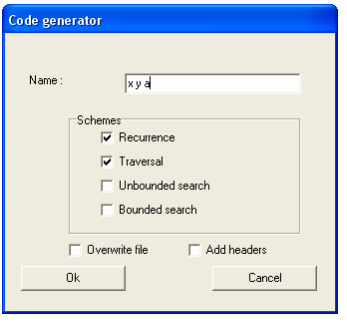


Fig. 2. Asking for an implementation

Finally, we show the generated code for the Java syntax.

```
int i;
float x0, x1, y0, y1, a0, a1;
x0=0; y0=0; a0=0;
i=0;
while (i<N) {
     i=i+1;
     x1 =x0 + d0*Math.cos(a0);
     y1 =y0 + d0*Math.sin(a0);
     a1 =a0 + 2*Math.PI/N;
     x0=x1; y0=y1; a0=a1;
}
```

3.2 Example 2. Carrying Different Weights

A worker is carrying different objects with different weights between two points. The first time he covers the distance, he carries $A>0$ units of weight. The second time, he carries $B>0$ ($A>B$) units of weight. As time goes by, his tiredness increases and he is forced to reduce the weight he can carry, which becomes the minimum between 95% of the last covered distance and 90% of the last distance but one. We pose the problem of determining the number of complete ways the worker can carry out before

he is exhausted, which occurs when the total carried weight exceeds *C*. We define a recurrence relation w_i to express the current weight, by the following equations:

$$w_0=A,\ w_1=B$$

$$w_i=minimum(0.95*w_{i-1},\ 0.90*w_{i-2}),\quad i>1$$

Notice that students must use a definition by cases in order to properly provide the tool with this equation. Thus, condition $0.90*w_0>0.95*w_1$ has to be supplied in the corresponding dialog box (Fig. 3).

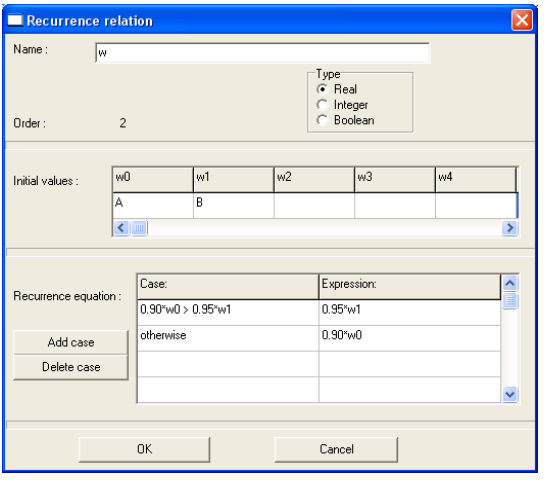


Fig. 3. Defining the recurrence relation w

Since we look for the first time the total carried weight exceeds *C*, we introduce a recurrence relation ac_i to hold the total carried weight:

$$ac_0=A,\ ac_1=A+B,$$
$$ac_i=ac_{i-1}+w_i,\ i>1$$

Observe that ac_i depends on w_i, thus the generated code must progress on the recurrence w before progressing on ac. The tool warns the student about such situation, and supplies a right ordering when required (Fig. 4 on the left).

Students must apply the unbounded search schema, thus the search condition has to be provided before generating the code (Fig. 4 on the right). Since we use the expression ac1>C, the value i-1 will finally return the number of complete ways before the worker becomes exhausted.

Fig. 4. Left: Advising that a proper ordering is required. Right: Asking for the search condition.

Then the generated Java code is the following:

```
int i; boolean found;
float w0, w1, w2, ac0, ac1, ac2;
w0=A; w1=B; ac0=A; ac1=A+B;
i=1;
found=ac1>C;
while (!found) {
  i=i+1;
  if (0.90*w0 > 0.95*w1) w2= 0.95*w1;
  else w2= 0.90*w0;
  ac2=ac1 + w2;
  w0= w1; w1= w2; ac0= ac1; ac1= ac2;
  found= ac1>C;
}
```

3.3 Integrating the Tool into the Course

We present the tool in the classroom after teaching the schemas for recurrence relations, which usually takes place at the end of the first out of two trimesters. Thus, the students have already been taught about their properties and about how schemas must be applied to specific problems. We have also presented some variations of the schemas (e.g. loops controlled by a counter or by a boolean expression), which are compared each other in order to gain the insights on them.

The tool is introduced to solve some of the problems they have manually coded previously. Students get really surprised when they notice that the tool solves the problems instantaneously. For the instructor, the tool can be used to prove that schema application is a real systematic process.

One hour is basically enough to explain the tool features. Next, the tool is uploaded to the web to make it public. Then students are encouraged to autonomously apply the tool to a selection of problems, as an optional lab assignment.

4 Evaluation of Schemas and the Tool

4.1 Study Framework

For the last two years, we have been analyzing the influence of using schemas on the development of students' programming skills, when teaching an introductory programming course during the first year of a Software Engineering degree, which applies the usual CS-first approach [12]. The study population was integrated in 7 groups each year, of around 70 students each; some of the groups (3 the first year, and 1 the second) studied schemas, while the others studied in a traditional language-oriented approach. Since students are randomly assigned to the groups, the groups are comparable with respect to the students' programming capabilities. We begin comparing the two approaches according to the academic success of the students, not considering other factors as the diversity of teachers and exams.

Only a few students decided to try CGR, despite of the extra mark that had been added to their final grade in case they would have solved some of the problems posed

in the lab assignment. Although they were not forced to solve all of them, they were asked to apply each of the three schemas at least once; hence, they should classify the chosen problems before trying to solve them, exploiting one of the schemas benefits. The assignment also included a satisfaction survey that students should fill in order to evaluate the tool.

4.2 Discussion about Academic Success

Table 1 reports the pass rates of the seven groups. The schemas-groups rates are displayed in boldface. The table points out that the schemas rates occupied the highest places, especially in 2007-2008.

Table 1. Pass rates

Group	1	2	3	4	5	6	7
2006-2007	31.7	**51.2**	51.4	**49.1**	**47.8**	27.7	39.7
2007-2008	29.7	**43.3**	25.9	25.9	23.5	40.2	51.5

4.3 Discussion about How Students Used the Tool

The programming assignment consisted of a list of 10 problems: six of them required the unbounded search schema (US), two the bounded search one (BS), and two the traversal one (T). Each student had to solve from 3 to 5 problems. The number of students that participated was 24 in 2007-2008. Table 2 shows the results we obtained. For each problem, we have studied four variables, which have been displayed in rows: the schema solving the problem (A), the number of students that chose the schema rightly|wrongly (B), the number of students that defined the recurrences rightly|wrongly (C), and the number of students that defined the condition of the (unbounded search or bounded search) schema rightly|wrongly (D).

Table 2. Report on the students' solutions

	1	2	3	4	5	6	7	8	9	10
A	US	US	BS	US	US	T	BS	T	US	US
B	1|0	2|0	11|9	6|2	12|3	19|0	11|0	12|0	4|0	14|0
C	1|0	2|0	16|3	8|0	13|2	12|6	10|0	9|3	2|1	13|1
D	1|0	2|0	5|5	5|2	10|3		10|0		2|1	14|0

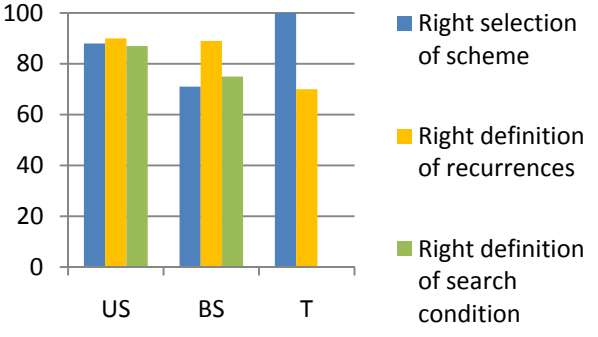
Table 3 summarizes the previous results, showing the success rates related to variables B, C and D for each schema. The average of these rates for the three schemas is 87% for the variable B, 84% for C, and 83% for D.

Thus, a descriptive analysis of the results shows that students usually choose the right schema, define the recurrence relations properly, and provide the tool with the correct search condition.

In order to analyze Table 3 more deeply, we have compared the percentages of the three schemas by pairs [11]. This study reports that the schemas US versus T, and BS

versus T, are different at the 95.0% confidence level (P<0.05), regarding variable B. Actually we conclude that problems based on traversals are more easily guessed, since the success rate for this schema is much higher than for the search ones.

Table 3. Analysis of students' solutions



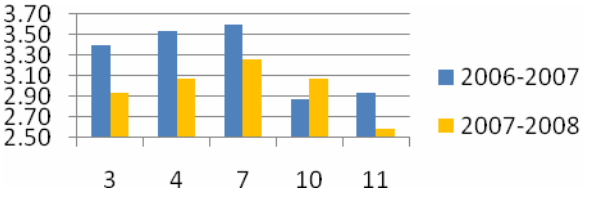
4.4 Discussion about Students' Compliance

The satisfaction survey was composed of 14 assertions. Students' answers ranged from 1 [total disagreement] to 5 [total agreement]. We focus on the most relevant questions:

3. "To define the relations involved in is an easier task than to code from scratch".
4. "To find out the relations takes less time than to code from scratch".
7. "The use of the CGR tool is suitable to solve problems in a first-year programming course".
10. "The CGR tool is useful to teach a first-year programming course".
11. "The CGR tool is a helpful tool to autonomous learning".
12. "Your programming background was broad before starting the course"

Table 4 shows the averaged answers to the previous questions in the two years. Notice that almost all of them exceed the middle value (3), thus students seem to be grateful for using the tool. We especially appreciate answers to question 3, 4 and 7, since they indicate that students find the tool useful to build programs.

Table 4. Students' answers to the selected assertions



In order to know whether the tool can be considered a useful tool we have analyzed the answers of 2007-2008 in depth. Concretely, we have studied whether any previous programming background affects the answers. Thus, we have applied the χ^2 test between answers to questions 3, 4, 7, 10 and 11 versus answers to question 12. Since the range of answers to these questions was too wide for 24 surveys, we have grouped the answers in order to safely apply this test. Thus, answers 1 and 2 have been replaced with 1, 4 and 5 with 5, and answer 3 has been ruled out. Then we have finally applied the independence tests over five 2x2 matrices. The study only reveals that answers to question 7 and 12 are not independent at the 95.0% confidence level ($P<0.05$) [10]. Concretely, on the one hand, 80.0% of the students with a broad background answered 1 to question 7, while 20.0% of them answered 5; on the other hand, 21.4% of the students with a narrow background answered 1 to question 7, while 78.6% of them answered 5. In consequence, we can conclude that the more previous background, the less they think that the tool is helpful to solve problems. In our opinion, students with previous programming skills do not like the tool because they feel uncomfortable when they cannot appeal to their own programming schemas.

5 Conclusions

Methodologies based on schemas are becoming popular for teaching programming in introductory courses. They focus on algorithm design instead of the language syntax, and use schemas as algorithm cognitive units. In this paper, we have presented a tool for programming using schemas. In order to solve a given programming problem, the student defines a recurrence relation system, selects the proper schema and the tool automatically generates the code that solves the problem in the target language. In this way, our tool allows the integration of methodologies based on schemas into the subject of the course.

We have also experimentally studied the influence of using schemas on the development of students' programming skills, and we have analyzed the students' compliance with the tool. The results we have obtained reveal that students assimilate schemas well. Regarding the tool, students find it suitable to solve problems and helpful to autonomous learning.

References

1. Abelson, H., di Sessa, A.A.: Turtle Geometry. MIT Press, Cambridge (1981)
2. Calloni, B., Bagert, D.: Iconic Programming Proves Effective for Teaching the First Year Programming Sequence. In: SIGCSE 1997, pp. 262–266. ACM Press, New York (1997)
3. Carlisle, M., Wilson, T., Humphries, J., Hadfield, S.: RAPTOR: A Visual Programming Environment for Teaching Algorithmic Problem Solving. In: SIGCSE 2005, pp. 176–180. ACM Press, New York (2005)
4. Grimaldi, R.P.: Discrete and Combinatorial Mathematics. Addison Wesley, Reading (2003)
5. Muller, O.: Pattern Oriented Instruction and the Enhancement of Analogical Reasoning. In: ICER 2005, Seatle, Washington, USA (2005)

6. Muller, O., Haberman, B., Ginat, D.: Pattern-Oriented Instruction and its Influence on Problem Decomposition and Solution Construction. In: ITiCSE 2007, pp. 151–155. ACM Press, New York (2007)
7. Odifreddi, P.G.: Classical Recursion Theory. North Holland, Amsterdam (1992)
8. Scholl, P.C., Peyrin, J.P.: Schémas Algorithmiques Fondamentaux. Séquences et iteration. Masson (1991)
9. Soloway, E.: Learning to program = learning to construct mechanisms and explanations. Comm. ACM 29(9), 850–858 (1986)
10. SPSS v.15, SPSS Inc. (1989-2006), `http://www.spss.com`
11. STATISTICA v. 7.1. SatSoft, Inc. (2005), `http://www.statsoft.com`
12. The Joint Task Force for Computing Curricula. Software Engineering 2004 (August 2004)
13. Watts, T.: The SFC Editor: A Graphical Tool for Algorithm Development. JCSC 20(1), 73–85 (2004)

The New Computational and Data Sciences Undergraduate Program at George Mason University

Kirk Borne, John Wallin, and Robert Weigel

Computational and Data Sciences, George Mason University,
Fairfax, VA 22030, USA

Abstract. We describe the new undergraduate science degree program in Computational and Data Sciences (CDS) at George Mason University (Mason), which began offering courses for both major (B.S.) and minor degrees in Spring 2008. The overarching theme and goal of the program are to train the next-generation scientists in the tools and techniques of *cyber-enabled science (e-Science)* to prepare them to confront the emerging petascale challenges of data-intensive science. The Mason CDS program has a significantly stronger focus on data-oriented approaches to science than do most computational science and engineering programs. The program has been designed specifically to focus both on simulation (Computational Science) and on data-intensive applications (Data Science). New courses include Introduction to Computational & Data Sciences, Scientific Data and Databases, Scientific Data & Information Visualization, Scientific Data Mining, and Scientific Modeling & Simulation. This is an *interdisciplinary science* program, drawing examples, classroom materials, and student activities from a broad range of physical and biological sciences. We will describe some of the motivations and early results from the program[1]. More information is available at http://cds.gmu.edu/.

1 Data-Intensive Science: A New Vision for Science Education

The development of models to describe and understand scientific phenomena has historically proceeded at a pace driven by new data. The more we know, the more we are driven to tweak or to revolutionize our models, thereby advancing our scientific understanding. This data-driven modeling and discovery linkage has entered a new paradigm [1]. The acquisition of scientific data in all disciplines is now accelerating and causing a nearly insurmountable data avalanche [2]. In astronomy in particular, rapid advances in three technology areas (telescopes, detectors, and computation) have continued unabated [3] – all of these advances

[1] The development of the Mason Computational and Data Sciences undergraduate program is sponsored by the NSF CCLI (Course, Curriculum, and Laboratory Improvement) program, through award # 0737091.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 74–83, 2009.

lead to more and more data [4]. With this accelerated advance in data generation capabilities over the coming years, we will require an increasingly skilled workforce in the areas of computational and data sciences in order to confront these challenges. Such skills are more critical than ever since modern science, which has always been data-driven, will become even more data-intensive in the coming decade [4,5]. Increasingly sophisticated computational and data science approaches will be required to discover the wealth of new scientific knowledge hidden within these new massive scientific data collections [6,7].

We live in an information age in which we are inundated with facts, tending toward information overload. Though the data glut problem is not limited to science, science is first and foremost a forensic discipline – we gather evidence, first to develop a hypothesis, then to test our hypothesis, and finally to vindicate or else to invalidate the hypothesis, at which point we gather more evidence, and the process continues. We must educate the next generation scientists, if not all citizens, in the principles of evidence-based reasoning, fact-based induction, and data-oriented science. In particular, we must muster educational resources to train a skilled data-savvy workforce: one that knows how to find facts (i.e., data, or evidence), access them, assess them, organize them, synthesize them, look at them critically, mine them, and analyze them.

2 Background and Motivation

The growth of data volumes in nearly all scientific disciplines, business sectors, and federal agencies is reaching epidemic proportions. This epidemic is characterized roughly by a doubling of data each year. It has been said that "while data doubles every year, useful information seems to be decreasing" [8], and "there is a growing gap between the generation of data and our understanding of it" [9]. In an information society with an increasingly knowledge-based economy, it is imperative that the workforce of today and especially tomorrow be equipped to understand data. This understanding includes knowing how to access, retrieve, interpret, analyze, mine, and integrate data from disparate sources. This is emphatically true in the sciences. The nature of scientific instrumentation, which is becoming more microprocessor-based, is that the scale of data-capturing capabilities grows at least as fast as the underlying computational-based measurement system[10]. For example, in astronomy, the fast growth in CCD detector size and sensitivity has seen the average size of a typical large astronomy sky survey project grow from hundreds of gigabytes 10 years ago (e.g., the MACHO survey), to tens of terabytes today (e.g., 2MASS[2] and Sloan Digital Sky Survey[3] [3], up to a projected size of tens of petabytes 10 years from now (e.g., LSST, the Large Synoptic Survey Telescope[4] [4]). LSST will produce one 56K x 56K (3-Gigapixel) image of the sky every 20 seconds, generating nearly 30 TB of

[2] http://www.ipac.caltech.edu/2mass/
[3] http://www.sdss.org
[4] http://www.lsst.org

data daily for 10 years. In the field of Space Weather and Solar Physics, NASA announced in 2008 a science data center specifically for the SDO (Solar Dynamics Observatory). The SDO will obtain one 4K x 4K solar image every 10 seconds, generating 1 TB of data per day. NASA recognizes that previous approaches to scientific data management, analysis, and mining will simply not work. Consequently, we see the floodgates of data opening wide in astronomy, high-energy physics, bioinformatics, numerical simulation research, geosciences, climate monitoring and modeling, and more. Outside of the sciences, it is widely documented that the data flood is in full force in banking, healthcare, homeland security, drug discovery, medical research, insurance, and (as we all have seen) e-mail. The application of data mining, knowledge discovery, text mining, and e-discovery tools to these growing data repositories is essential to the success of agencies, economies, and scientific disciplines.

2.1 Data Sciences: A National Imperative

The article "Agencies Join Forces to Share Data" calls for more training in data skills [11]. This article describes a new Interagency Working Group on Digital Data, representing 22 federal agencies in the U.S., including the NSF, NASA, DOE, and more. The group plans to set up a robust public infrastructure so that all researchers have a permanent home for their data. One option is to create a national network of online data repositories, funded by the government and staffed by dedicated computing and archiving professionals. Who are these computing and archiving professionals? We believe that this professional workforce must be trained in the disciplines of computational and data sciences. We are addressing this societal need through the new CDS curriculum in the Mason Department of Computational and Data Sciences (CDS).

Within the scientific domain, Data Sciences is becoming a recognized academic discipline. In a recent Data Sciences Journal article [12], it is argued that now is the time for Data Sciences curricula. In another article [13], Data Science is again promoted as a rigorous academic discipline. Further, there was a 2007 NSF-cosponsored workshop on Data Repositories, which included a track on data-centric scholarship, where they explicitly state what we now believe: "Data-driven science is becoming a new scientific paradigm – ranking with theory, experimentation, and computational science" [14]. Another Data Sciences Journal article states: "Without proper management of continuously-produced important data and without the productivity of new disciplines based on data, we cannot solve important problems of the world" [15]. An excellent article recently described Informatics, the new paradigm of data-intensive science, as "the use of digital data, information, and related services for research and knowledge generation" [16]. Consequently, many scientific disciplines are developing subdisciplines that are information-rich and data-based, to such an extent that these are now becoming (or have already become) recognized stand-alone research disciplines and academic programs on their own merits. The latter include bioinformatics and geo-informatics, but will soon include astroinformatics, e-Science, medical informatics, and data science. Several national study groups have issued reports

on the urgency of establishing scientific and educational programs to face the data flood challenges:

1. National Academy of Sciences report: "Bits of Power: Issues in Global Access to Scientific Data" (1997) [17];
2. NSF report on "Knowledge Lost in Information: Report of the NSF Workshop on Research Directions for Digital Libraries" (2003) [18];
3. NSB (National Science Board) report on "Long-lived Digital Data Collections: Enabling Research and Education in the 21st Century" (2005);
4. NSF "Atkins Report" on "Revolutionizing Science and Engineering Through Cyberinfrastructure: Report of the National Science Foundation Blue-Ribbon Advisory Panel on Cyberinfrastructure" (2005) [19];
5. NSF-sponsored report with the Computing Research Association on "Cyberinfrastructure for Education and Learning for the Future: A Vision and Research Agenda" (2005);
6. NSF report on "Cyberinfrastructure Vision for 21st Century Discovery" (2007) [20];
7. JISC/NSF Workshop on Data-Driven Science & Repositories (2007) [14].

Each of these reports issues a call to action in response to the data avalanche in science, engineering, and the global scholarly environment. For example, the NAS "Bits of Power" report lists 5 major recommendations, one of which includes: "Improve science education in the area of scientific data management" [17]. More recently, the Atkins Report stated that skills in digital libraries, metadata standards, digital classification, and data mining are critical [19].

3 Computational and Data Sciences at Mason: CUPIDS

The new Computational & Data Sciences curriculum at Mason uniquely responds to the recommendations of these national studies and reports. The urgent need for such a curriculum cannot be overstated, as the Atkins Report has said: "The importance of data in science and engineering continues on a path of exponential growth; some even assert that the leading science driver of high-end computing will soon be data rather than processing cycles. Thus it is crucial to provide major new resources for handling and understanding data" [19]. The core and most basic resource is the human expert, trained in key data science skills. As stated in the 2003 NSF "Knowledge Lost in Information" report, human cognition and human capabilities are fundamental to successful leveraging of cyberinfrastructure, digital libraries, and national data resources [18].

CUPIDS[5] is the NSF-supported "Curriculum for an Undergraduate Program in Data Sciences" at Mason in the CDS Department. The central goal for the CUPIDS project is *to increase student's understanding of the role that data plays across the sciences as well as to increase the student's ability to use the technologies associated with data acquisition, mining, analysis, and visualization.* We have five objectives for this project:

[5] Funded through NSF Award # 0737091.

1. To teach students what Data Science is and how it is changing the way science is being done across the disciplines
2. To change student's attitudes about and improve their confidence in using computers to address scientific data problems
3. To increase student's abilities to use visualization for generating and addressing scientific questions
4. To increase student's abilities to use databases for scientific inquiry
5. To increase student's abilities to acquire, process, and explore experimental data with the use of a computer

As evident from this list, the goals of the CUPIDS project are focused primarily on the Data Sciences, which are a subset of the educational goals and programs within CDS. We describe several aspects of these programs below.

3.1 The CDS Degree Program and Curriculum

Students are required to complete a total of 18 credits (6 core courses) in computational and data sciences (CDS), 15 credits in computer science, 23 credits in mathematics, 6 credits in statistics, 21-25 credits in a science concentration, and 3-9 credits in CDS electives. Three concentration areas are currently offered: Physics, Chemistry, and Biology. Additional concentrations may be added to the program in the future (perhaps astronomy, materials science, and geosciences). For a given concentration, the 21-25 credits that a student must take in that science discipline include the core courses that are required for majors matriculating in those programs. As much as possible, the core CDS courses include scientific examples and applications from all of the science concentrations. Of course, this implies a certain degree of heterogeneity in the scientific knowledge of the students who come from different fields. Consequently, the primary focus of the CDS courses are on the techniques of computational and data sciences, and not on the specific experimental and theoretical bases of the various science disciplines. As part of their education, students are encouraged to undertake an optional research project that allows them to gain useful experience in the development of simulations and other aspects of computational science. To facilitate this interdisciplinary science environment, the CDS faculty have degrees in the science disciplines (including Astronomy & Astrophysics, Space Weather, Physics, Computational Fluids, Materials Science, Computational Chemistry, Computational Statistics, Applied Mathematics, and more) and many of the faculty have affiliations (or joint appointments) within those other departments.

The six required computational and data sciences core courses are:

1. *CDS 101 Introduction to Computational & Data Sciences* - Introduces the use of computers in scientific discovery via simulations and data analysis.
2. *CDS 301 Scientific Information and Data Visualization* - The techniques and software used to visualize scientific simulations, complex information, and data visualization for knowledge discovery.
3. *CDS 302 Scientific Data and Databases* - Data and databases used by scientists, including data types, database queries, and distributed data systems.

4. *CDS 401 Scientific Data Mining* - Data mining techniques from statistics, machine learning, and visualization applied to scientific knowledge discovery.
5. *CDS 410 Modeling and Simulations I* - Numerical differentiation and integration, initial-value and boundary-value problems for ordinary differential equations, methods of solution of partial differential equations, iterative methods of solution of nonlinear systems, and approximation theory.
6. *CDS 411 Modeling and Simulation II* - The application of modeling and simulation methods to various scientific applications, including fluid dynamics, solid mechanics, materials science, molecular mechanics, and astrophysics.

We describe below two of the core courses in more detail and we enumerate the desirable skills that we expect for students who complete the program of study.

3.2 Course: Introduction to Computational and Data Sciences

This course provides an interdisciplinary introduction to the tools, techniques, methods, and cutting edge results from across the Computational and Data Sciences. Students are shown how computational tools are fundamentally changing our approach in the experimental, observational, and theoretical sciences through the use of data and modeling systems. No mathematical background is assumed, other than high school algebra. Qualitative results are emphasized, to show the problems and challenges facing researchers today. Examples are drawn from both the "real world" familiar to students and also from the frontiers of science where these techniques are being used to solve complex problems.

Upon completion of the course, students should be able to:

1. Describe how data are represented within a computer, from binary numbers to arrays and databases.
2. Explain how scientific data are acquired, processed, stored, reduced, and analyzed using computers.
3. Express how we create knowledge from data and information using visualization and data mining.
4. Create effective ways to visualize simple data sets.
5. Conduct and explain simple simulations of complex phenomena.
6. Express how changing computing technologies further scientific research, and how the technological and scientific progress are tied together.

Lecture topics in the course include: the scientific method; computer internals (binary numbers and logic circuits); computer algorithms and tools (Matlab introduction); data acquisition; signal processing (understanding noise and error); scientific databases; data reduction and analysis; data mining; computer modeling; numerical simulations; visualization; high-performance computing; and future directions in computational science.

3.3 Course: Scientific Data Mining

This course provides a broad overview of the knowledge discovery (data mining) process, as applied to scientific research. Data mining is the search for

hidden meaningful patterns in large databases (e.g., *find the one gene sequence in a large genome DNA database that always associates with a specific cancer*). These patterns and relationships are often expressed as rules (e.g., *if a blue star-like object is found next to a faint unusual-shaped galaxy in a large astronomy database, then the blue object might be a distant quasar whose outburst in being triggered by a collision with that galaxy*). Consequently, data mining is sometimes referred to as the process of converting information from a database format into a knowledge-based rule format. Identifying these patterns and rules from enormous data repositories can provide significant competitive advantage to scientific research projects and in other career settings.

Data mining is motivated and analyzed in this course as the "killer app" for large scientific databases (i.e., a key enabler of scientific discovery). Data mining techniques, algorithms, and applications are covered, as well as the key concepts of machine learning, data types, noise handling, feature selection, data transformation, and similarity/distance metrics. Techniques are analyzed specifically in terms of their application to scientific research problems. Several scientific case studies are drawn from the science research literature, including astronomy, space weather, geosciences, climatology, bioinformatics, numerical simulation research, drug discovery, health informatics, combinatorial chemistry, digital libraries, and virtual observatories. Prerequisites for this course include the undergraduate Scientific Data and Databases course and mathematics/statistics courses.

Upon completion of the course, students should be able to:

1. Express the role of data mining within scientific knowledge discovery.
2. Express the most well known data mining algorithms and correctly use data mining terminology.
3. Express the application of statistics, similarity measures, and indexing to data mining tasks.
4. Identify appropriate techniques for classification and clustering applications.
5. Determine approaches used for mining large scientific databases (e.g., genomics, virtual observatories).
6. Recognize techniques used for spatial and temporal data mining applications.
7. Express the steps in a data mining project (e.g., cleaning, transforming, indexing, mining, analysis).
8. Analyze classic examples of data mining and their techniques.
9. Effectively prepare data for mining and use data mining software packages.

Lecture topics in the course include: scientific motivation for data mining; quantitative and statistical concepts; software packages; data preparation (previewing, cleaning dirty data, normalization, transformation); distance and similarity metrics for clustering and classification; supervised learning methods; unsupervised learning methods; scientific data mining case studies; and special topics (time series, image mining, spatial data, and outlier/event/anomaly detection).

3.4 Results from the Program and Future Work

Early results from the program include the following: (a) the B.S. (CDS major) was approved in 2007; (b) the minor was approved in 2008; (c) the first courses

(CDS 101 and 302) were offered in Spring 2008; (d) additional courses (CDS 301 and 401) were offered in Fall 2008; (e) ∼10 students are currently majoring in the program; and (f) one new course has been approved and will be offered in the coming year: *CDS 151 Data Ethics in an Information Society.*

We have identified a need for additional courses in order to retain students in the program. The initial selection of core courses includes only one course before the Junior year: CDS 101. This was not a problem with the first wave of students majoring in the program, all of whom were Junior-level transfer students. To address the retention problem for the newest students, we are developing additional courses, which may include courses in computational tools for scientists and discipline-specific topics. We have received a small "pedagogy" grant from the Mason College of Science to develop a new Gen Ed course for science majors: *CDS 120 Computational and Data Tools for Scientists.* This course will employ novel student-led peer instruction approaches, to enable the course to scale to large numbers of students. The goal is to make this course the default computational tools course for all science majors at Mason. At present, these students take a computer languages course from the I.T. school, which does not have a science focus nor does it include scientific applications. CDS 120 will cover presentation tools (Powerpoint, HTML), analysis tools (spreadsheets), databases, search methods, basic programming in Matlab, overview of data acquisition and signal processing, and numerical simulations (verification and validation).

In the area of discipline-specific courses, we are considering developing a basic course in tools for computational biology and bioinformatics, to meet the specific needs of students in Biology. Though this course is biology-specific, it is consistent with another discipline-specific course that is now under review: "Materials Science with Applications to Renewable Energy." We are also discussing with the Mason Physics & Astronomy department a concept for developing a course in Astroinformatics. In these cases, we are drawing upon the considerable expertise of the CDS faculty in specific science areas to develop and to offer computational and data science courses for majors in those scientific disciplines. Such courses would not be required for all CDS majors, but they will be appropriate electives for CDS students in the corresponding science concentration.

3.5 Similar Programs at Other Universities

We are aware of a few similar programs at other universities. In nearly all of these cases, the focus is either on (i) computational science (with little attention to data sciences), (ii) data sciences (generically, not within the context of teaching science), or (iii) data sciences within the context of a single specific science. Type (i) progams include the CACR (Center for Advanced Computation and Research) at Caltech (who do have a strong connection with Astronomy and therefore are moving toward a focus on cyber-enabled data sciences), and many other computational science programs (e.g., LSU's CCT, U. Texas' TACC). Type (ii) programs include the Discovery Informatics program at the College of Charleston. Type (iii) programs include the POCA (Partnership in Observational and Computational Astronomy) at SCSU and Clemson University, Purdue's Discovery Informatics

program, the emerging joint programs between CS and astronomy departments at Notre Dame, and similarly at U. Michigan. Beyond these are the science programs in data sciences, including Cornell's new DISCOVER data-driven science program[6] (with a focus on astronomy plus other disciplines) and the new e-Science Institute at U. Washington[7] (focusing on oceanography, environmental sciences, and astronomy) – these two programs appear to be most similar to the Mason CDS program.

4 Conclusions and Summary Remarks

Computational and Data Sciences are emerging fields involving applications of sophisticated simulation and data-oriented methods to build models and solve problems in science and engineering. Recently emerged interdisciplinary areas in the chemical, physical and biological sciences (such as biotechnology, nanotechnology, molecular electronics, photonics in nanoscale systems, and energetics of DNA/protein binding) require highly-qualified professionals with strong computational skills to work closely with experimentalists in solving complex scientific and engineering problems. Emerging data-intensive science fields (such as Geoinformatics, Bioinformatics, Astroinformatics, and Materials Informatics) require specially trained professionals with strong data skills to address ubiquitous data-intensive applications in science, industry, and government. Computational and data sciences complement existing theoretical and experimental science approaches and may be thought of as a new mode of scientific inquiry.

The new CDS undergraduate *science* program at Mason complements the existing graduate program in Computational Science & Informatics (CSI), which has existed since 1992, having graduated 175 PhDs to-date. There are over 90 graduate courses in the Mason CSI program, covering many science disciplines. In conclusion, we summarize the key features of the CDS program:

- *Who?* – Students with a broad interest in computers and sciences will benefit from the program.
- *Why?* – Students graduating with a traditional discipline-based bachelors degree in biology, chemistry, or physics generally do not have the required computational background necessary to participate as productive members of modern interdisciplinary scientific research teams, which are becoming increasingly computational- and data-intensive. The motivating theme and goal of the CDS program are to train the next-generation scientists in the tools and techniques of *cyber-enabled science (e-Science)* to prepare them to confront the emerging petascale challenges of data-intensive science.
- *What?* – The BS program in CDS provides science students with a variety of opportunities to become research professionals possessing interdisciplinary knowledge, including sciences and applied mathematics, augmented with strong computational and data-oriented skills. This program has a significantly stronger focus on data-oriented approaches to science than do most

[6] http://arecibo.tc.cornell.edu/DRSG/Links.aspx
[7] http://escience.washington.edu/

Computational Science and Engineering (CSE) programs. Graduates from this program will acquire interdisciplinary knowledge and will be able to apply scientific principles in solving complex real-world problems.

– *How?* – Students in this program are exposed to a wide range of computational and data science applications, and will learn computational science tools, high-performance computing, applied and theoretical computational techniques, modeling and simulation, statistical analysis, optimization, data & information visualization, scientific database applications, scientific data mining & knowledge discovery in databases (KDD), and data-intensive science research methods. The CDS program has been designed specifically to focus both on simulation (Computational Science) and on data-intensive applications (Data Science) within an interdisciplinary science environment.

References

1. Mahootian, F., Eastman, T.: Complementary Frameworks of Scientific Inquiry. World Futures journal (2009) (in press)
2. Bell, G., Gray, J., Szalay, A.: arxiv.org/abs/cs/0701165 (2005)
3. Gray, J., Szalay, A.: Microsoft technical report MSR-TR-2004-110 (2004)
4. Becla, J., et al.: arxiv.org/abs/cs/0604112 (2006)
5. Szalay, A.S., Gray, J., VandenBerg, J.: arxiv.org/abs/cs/0208013 (2002)
6. Gray, J., et al.: arxiv.org/abs/cs/0202014 (2002)
7. Borne, K.D.: Data-Driven Discovery through e-Science Technologies. In: 2nd IEEE Conference on Space Mission Challenges for Information Technology (2006)
8. Dunham, M.: Data Mining Introductory and Advanced Topics. Prentice-Hall, New Jersey (2002)
9. Witten, I., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco (2005)
10. Gray, J., et al.: Scientific Data Management in the Coming Decade, arxiv.org/abs/cs/0502008 (2005)
11. Butler, D.: Agencies Join Forces to Share Data. Nature 446, 354 (2007)
12. Smith, F.: Data Science as an Academic Discipline. Data Science Journal 5, 163 (2006)
13. Cleveland, W.S.: Data Science: an Action Plan for Expanding the Technical Areas of the Field of Statistics. International Statistics Review 69, 21 (2007)
14. NSF/JISC Repositories Workshop (2007), `http://www.sis.pitt.edu/~repwkshop/`
15. Iwata, S.: Scientific "Agenda" of Data Science. Data Science Journal 7, 54 (2008)
16. Baker, D.N.: Informatics and the 2007-2008 Electronic Geophysical Year. EOS 89, 485 (2008)
17. Bits of Power: Issues in Global Access to Scientific Data, `http://www.nap.edu/catalog.php?record_id=5504`
18. Knowledge Lost in Information: Report of the NSF Workshop on Research Directions for Digital Libraries, `http://www.sis.pitt.edu/~dlwkshop/report.pdf`
19. Report of the NSF Blue-Ribbon Advisory Panel on Cyberinfrastructure, `http://www.nsf.gov/od/oci/reports/atkins.pdf`
20. Cyberinfrastructure Vision for 21st Century Discovery, `http://www.nsf.gov/pubs/2007/nsf0728/index.jsp`

Models as Arguments: An Approach to Computational Science Education

D.E. Stevenson

315 McAdams Hall, School of Computing, Clemson University,
PO Box 34097, Clemson, SC 29634-0974
steve@cs.clemson.edu

Abstract. Hardware and software technology have outpaced our ability to develop models and simulations that can utilize them. Furthermore, as models move further into unfamiliar territory, the issues of correctness becomes more difficult to assess. We propose extending classical argumentation structures as the basis for computational science education.

1 Human Centric Computing — The Argument

Computation and computational science has rightly emphasized hardware and software development. But the hardware is now far more advanced than the software developers can take advantage of and the scientific problems are more complex than those of fifty years ago. Now that computing is pervasive (if not invasive) it is time for humans to catch up. Kurtzweil's "singularity" not withstanding, we must modernize human interaction with computing. The futurists seem to be divided into two camps: pervasive computing and human centric computing (HCC). Once again, however, technology and not people are the focus, with overriding objectives of making the technology transparent *or* letting each sector focus on what it does best.

- The pervasive computing camp argues that IT needs to "get the end user experience right," focusing on designing devices that are easier for users to interact with in order to find relevant information.
- The objectives of human-centric computing are not to focus on the devices themselves, but rather to create an entire solution so that the human, rather than the device, is always connected.

This is all very exciting: but the human also needs a tune-up, meaning that the education of the 21st Century computational scientist cannot just be more detail at a higher level using the same techniques as those used fifty years ago. So far, the future is about more speed, not about more understanding. The fundamental question before computational science is still "How do we know it's right?"

Since it was fair to test machines for intelligence, machines might test humans with a variant of the Turing test that I will call a Kurzweil Test:

> A machine judge engages in a "machine language" conversation with one human and one machine, each of which try to appear as machines.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 84–92, 2009.

> All participants are placed in isolated locations. If the judge cannot reliably tell the human from the machine, the human is said to have passed the test. [1]

But we are before the singularity, so the Test is to determine if a machine is more intelligent than the human. What do we mean by "more intelligent"? I approach the test by using Bloom's Taxonomy (Fig. 1), which orders cognitive tasks from lowest to highest, as the measure of intellectual achievement and ask questions based on the standard verbs used to identify levels. Now we must ask, "Kurzweil machines will be able to pass the Turing test, will our students pass such a Kurzweil test?"

Remembering	define, recall, repeat, reproduce state
Understanding	classify, explain, identify, translate, paraphrase
Applying	choose, demonstrate, interpret, operate, solve, write.
Analyzing	appraise criticize, examine, experiment, question, test.
Evaluating	defend, judge, select, value, evaluate
Creating	construct, create, design, develop, formulate, write.

Fig. 1. Bloom's Taxonomy as modified by Lorin Anderson

We can hope that the Ph. D. students can pass a Turing test in their field, but undergraduates probably cannot. Firstly, there is a disconnect between student's abilities to respond to requests to develop artifacts and teachers' expectations. In learning to program, for example, we immediately ask students to **create** artifacts when their cognitive abilities are still in the lower three categories.

Secondly, the current idea that critical thinking is an ultimate skill to teach leaves us short of creating things: critical thinking is analyzing and evaluating, not creating. In fact, if one looks at classical rhetoric, one finds that critical thinking is a subordinate skill to argumentation.

We extend the concepts presented in [2]. Sect. 2 describes computational science problems as we propose they should be solved in a classroom. Sect. 3 discusses a pedagogical approach Systems-Questions-Explanations (SQE). Finally we introduce the addition of argumentation as a pedagogical approach to computational science (Sect. 4) followed by an example (Sect. 5). We collect our major points in Sect. 6.

2 The Problem

Computational science in its broadest sense is the modeling of systems. But system models, by themselves, are not necessarily correct. Aristotle was a prolific modeler whose models carried the force of law for almost 2,000 years — but he got the structures of both the universe and biology wrong. Francis Bacon argued in

[1] This tongue-in-cheek version is Wikipedia's explanation of the Turing test [1] with "human" and "machine" swapped.

his *Novum Organon* that models must be validated by experiments: the Scientific Method. It is now recognized that the Scientific Method is supported by a third leg: computing. Correctness is an ever more important now than with Bacon and the terms *verification* and *validation* are used to describe the processes. Recently, Wilson added *reproducibility* so we now abbreviate the "correctness" process as VV&R.

Despite our advanced computer systems, models and simulations are still developed by people. Traditional science and engineering education proceeds by showing the student a number of models, then asking the students to apply these models to textbook problems. This pedagogical approach assumes that students learn to model by exposure. Our data shows that the students are not learning how to create models.

Experiences with incoming science, technology, engineering, and science (STEM) students at Clemson indicates that students do indeed have the facts at hand that we suppose they should have. Many students have high SAT/ACT scores and high predicted GPAs; yet many never complete a STEM degree. One possibility is that these students do not have the critical thinking skills required to analyze scientific models; however, critical thinking skills testing we have conducted (with IRB approval) shows that these students seem to have the critical thinking skills of relatively mature thinkers [3]. In other words, we have students who are at the evaluting stage in Bloom's Taxonomy but still are not thriving.

The students seem not to understand how to structure an argument, nor are they able to evaluate their own arguments. As discussed in Parham, Chinn, and Stevenson [4], students have trouble solving unstructured problems that have many transitions among the six Bloom elements. By using verbal protocol approach, we were able to trace the students use of information and metacognition. Reviewing, analyzing, and evaluating student or historical arguments is one way to practice critical thinking. We can also practice argumentation by reviewing, analyzing, and evaluating arguments. But argumentation should also require students to structure an argument then review and self-evaluate it [5]. One approach we are trying is to organize modeling classes around classical argumentation methods. Argumentation is both a process (the discussion) and a product (the outcomes) consisting of three parts: rhetoric, reasoning and dialectic. Rhetoric means development of knowledge through communications and dialectic is discovering and testing knowledge through questions and answers. To conduct an orderly argument, we must share standards and appraisal methods relating to knowledge and the reasoning schemata.

3 How Do Humans Formulate Models? SQE

The first step in solving any problem is to construct a model of that problem, and this first model is a mental model. The same process of modeling is needed to install new ideas in our personal knowledge bases. The term *mental model* is used both in formally and informally. Many disciplines have their own interpretations of the term *model*, but we use three interpretations: (1) model of a physical

system (physical sciences), (2) models of axioms (logic), and (3) mental models (psychology). The basic interpretation of model is a set of logical (not necessarily classical logic) statements made about the modeled phenomenon that are jointly true — recall that mathematical equations are true or false. The processes for showing models (or simulations) are true are called verification and validation. The "subversive" purpose of this paper is to propose a practical approach to VV&R education.

While the results of a modeling study are external artifacts, the process of developing models is primarily psychological and sociological. Focusing too much on the external artifacts lends no insight into the thinking process: the reasoning process that education must hone, something quite different than having more theorems and laws. Questions concerning human reasoning are not new, but many modern studies started in the mid-1970s. Philip Johnson-Laird [6] based his concept of mental model on models of explanation put forth by Kenneth Craik [7]. Craik proposed a three step process: (S1) translation of original problem statement to an internal representation, (S2) the manipulation and reasoning to develop a solution, and finally (S3) the re-translation of the solution to an external medium.

We discuss S1 and S3 in this section and Step S2 is discussed in 4. Step S1 is a linguistic step, emphasizing two auxiliary steps: association of words with concepts (mind maps) and the association of concepts and relationships (concept maps). At the end of the translation step, the internal representation is a graph of schemata. Schemata are units of knowledge proposed by Immanuel Kant and finally popularized by John Anderson in the ACT* models. Work on schemata and their use in problem-solving in computer science is being studied at Clemson under the author. Step S3 is interesting in computational science because the output form is usually quite different than the input: complex natural language in, complex computer system out.

3.1 Psychology

The author began studying the human aspect of modeling and simulation approximately fifteen years ago, primarily studying problem-solving as described by Sternberg [8]. There are five aspects of problem solving: modeling (planning), reasoning, knowledge, critical thinking, and creative thinking. Bushey [3] explored the role of critical thinking, showing that successful computer scientists are critical thinkers. As research continued into the psychology literature, we soon focused on the issue of knowledge and how it is used.

Anderson [9] proposed the first of a series of models[2] collectively called ACT*, based on the concept that humans hold knowledge in certain memories, that these units of memories are connected, and that we use pattern-matching to find new connections. The dynamics of how this might occur were partially established by Marshall [10]. While Marshall established some evidence for schemata, she did not establish how we recognize mistakes. This process is now known

[2] These models are an active domain of research in cognitive psychology.

as metacognition, first introduced by Flavell [11]. The emerging picture is that the problem solver recognizes patterns of information that identifies schemata and their connections with metacognition playing a control function. Schemata themselves seem to be units of knowledge with ability to determine applicability, planning, and execution requirements.

4 Systems, Questions, and Explanations

This is still an unsatisfactory explanation of how humans learn to be modelers: how did Einstein happen on his elevator? Our goal is to understand the tools needed to model a system and inculcate them into the students. This was discussed in Stevenson [2], which proposes that the modeling is based on three foci: Systems, Questions, and Explanations (SQE). The reasoning for the SQE model is as follows: In order to ask a meaningful question, one must first have a system in mind. The system effectively defines the context, rules, and conditions that the system obeys and provides a vocabulary of variables and values (at resolution) to provide semantics for both the question and the answer. With the system and question, one can provide meaningful answers, which themselves must be explained; justification is just one aspect of explanation.

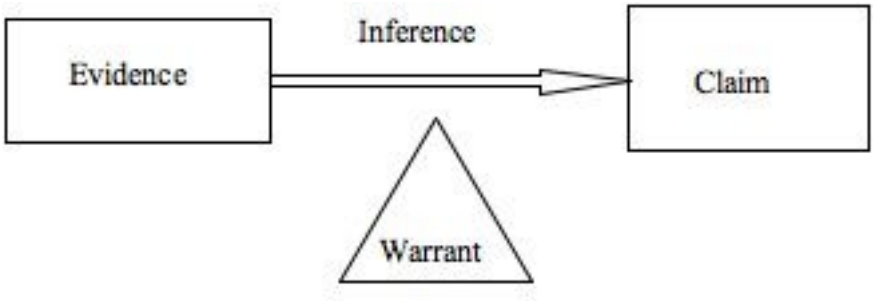


Fig. 2. Simplified Unit Argumentation Cell. Four fundamental information types. Claims are the conclusion. Evidence supports claims through inference. Inference is justified by warrants.

The development of an argument relies on the structure shown in Fig. 2. Classically, these parts are logical statements playing four roles:

1. Claim. This is the statement to be the conclusion.
2. Inference. A gap can be filled by inference based on the structure of the model. These inferences must be justified (warranted).
3. Evidence and Hypotheses. A gap may be filled by hypotheses that must then be proven.
4. Computation. A gap may imply enough conditions that a computation is feasible and practical.

This particular approach comes from a basic observation: one must argue that a model is correct and therefore the model is the argument. This model accepts the closed world view. In the open world models, there are defaults and uncontrolled logical situations; in the closed world, all propositions must be expressed explicitly. There are fine lines of distinction between claims, evidence, inference, and warrants that the student must wrestle with, but it is a clear framework.

Finally, the work on mental models [12,13] shows that most people use informal methods of reasoning [5]. Evidence indicates that we reason through logical models, but not necessarily using classical (simple syllogistic) logic. One approach, for example, might be to integrate new information in the older model, then deal with the contradictions and incoherencies. The basic problem is that classical logical deduction only works if we have all the facts and we have a set of axioms and we have a set of rules of inference. By definition, as we learn we do not have those things — they're a work in progress. This particular issue is now known as "informal logic".

4.1 Argumentation

Classically, there are three parts to discourse: logic, dialectic (discussing the truth of opinions) and rhetoric. A major theme in rhetoric is argumentation, including debates. [Note: There is no way to do justice to argumentation in such a short paper. We present a very short overview]. Arguments are decision-making processes that is carried out in an uncertain environment. An argument consists of four elements: claims, evidence, inference, and warrants, themselves logical statements. Claims are the conclusion of an argument and are not necessarily universal truths and hence the participants values, priorities, and methods of judgment are crucial. In order to arrive at a comon ground, the participants must adopt

- A critical thinking regimen,
- A common language and semantics,
- Procedural assumptions and norms
- A common frame or frames of reference

Fundamental reasoning skills are essential. The Unit Cell (Fig. 2) is effectively the rationale for accepting the claim when an inferential step should be accepted. Conclusions (claims) must be justified (warranted). Unlike formal syntactic justifications, arguments can be subjective [14], because knowledge can also be subjective. The adoption of skepticism is required, hence critical thinking is a key component. Knowledge has degrees of strength and is always provisional: our understanding could change with a simple experiment.

How can we use this for 21st Century education? The subject provides a framework to consider the development of a field. The following classical subjects form such a framework [15,16]:

- Evaluating Evidence. This is the study of both good and bad arguments. This is similar to requiring students to look at models and deciding whether they believe the evidence. The first third of [17] emphasizes these skills.

- Understanding the Issues. Students often do not know what the issue is when it comes to analyzing a problem. Most problems have a limited number of issues that must be addressed, but those issues must be apparent to the students.
- Case Construction. Case construction is the heart of argumentation, by this I mean choosing the unit cells (Fig. 2)you wish to use. Each subject has a limited number of standard arguments that any competent practioner understands.
- Inferences and Warrants. While most academic subjects try to use classical logic (But not intuitionistic logic ... grist for another article), the psychological studies of human reasoning indicate humans do not naturally do so. Formulating the statement is not the same as warranting (justifying) the statement. Physics principles such as Conservation of Energy laws fit in this area.
- Fallacious Reasoning. Fallacies take up a large portion of argumentation classes, but it is not clear how much time should be taken up in "applied" classes. Some disciplines such as mathematics have famous fallacies (the *pons asinorum*, for example).

Fortunately there are already examples of such approaches in computational science.

Test Driven Development (TDD). This is a method of developing software that adopts a test-first approach. Instead of writing a complete program first and then testing the program once it has been developed, TDD has the software programmers constantly testing the software from the first step. Tests are created for most or all of the individual elements of a program, such as the functions, objects, etc.

Model-Based Techniques. Model-based testing is closely related to model-based specification. Models are used to describe the behavior of the system under consideration and to guide such efforts as test selection and test results evaluation. Both testing and verification are used to validate models against the requirements and check that the implementation conforms to the specification model. Of particular importance are formal models with precise semantics, such as state-based formalisms. Techniques to support model-based testing are drawn from diverse areas, like formal verification, model checking, control and data flow analysis, grammar analysis, and Markov decision processes.

Patterns as Schemata. The formal descriptions of patterns, as in [18], are attempts to describe successful solutions to common problems. Patterns exist across the knowledge spectrum, in all disciplines. Software patterns are a type of schemata. Educating students to recognize patterns helps connect them with their own knowledge rather than just resolving a problem. Not only do patterns teach useful techniques, they help people communicate better, and they help people reason about what they do and why.

This is not very mystical: electrical engineering circuit diagrams are a type of pattern. Recognition of patterns as decomposition tools in problem solving is a hallmark of expertise. A person's collection of schemata concerning a problem form a graph representing that person's knowledge: a language about a particular program providing vocabulary for talking about a particular problem. Patterns have a context in which they apply.

5 An Example

An example of how this approach can be used actually occurred recently in one of my classes. This class has a semester-long project with seven milestones. Starting with Milestone 2, the students must test their own code before I do. One would think that seniors know how to think about testing such a project — unfortunately, ours do not. Therefore, the problem to be solved is "how to thinking about testing."

What is a model to the problem at hand? They want to claim that they have completely tested their code. I drew the unit cell on the board and labeled it. I then asked the students (I use problem-based learning exclusively), "What do you want to claim?" It took some time for them to respond, "We want you to accept our milestones as correct."

"What is your evidence"?

This took considerably longer before they could formulate that the evidence was actually four separate classes of processing. The project they were working on requires them to accept input or reject it. While they finally determined that there were four classes of input to demonstrate, they never did determine that they also had to show they should reject some inputs. When I pointed that out, we have eight separate types of test inputs.

"Why should I believe you"?

Again, this took some discussion, but we finally agreed that they were presenting direct evidence and could map the specification onto the inputs and outputs.

6 Conclusions

Early versions of this paper were laid out in a tableau format to simplify the task of organizing the material. An interesting observation is that the tableau is almost exactly what Euclid taught in the *Elements*. The connection is important, but our main message is this:

- To develop the best in our students they must reach the "creating" level of Bloom's Taxonomy as quickly as possible.
- To be at the creating level means that both critical thinking and argumentation (modeling) must be emphasized.
- Modeling begins with mental modeling.
- Modeling is an inductive process.

Final Comment. It may seem odd that there is little mention of *reasoning* and *logic*. There are several reasons for this, primarily the fact that any discussion of these subjects is a paper unto itself.

Acknowledgement

The author wishes to acknowledge the discussions with Dr. Donald Chinn, Ms. Jennifer Parham and the entire group working with the First Generation College Students at Clemson: Drs. Barbara Speziale, Jeffrey Appling, Calvin Williams, Robert Ballard, Matthew Ohland, and John Wagner. Ms. Sherry Dorris provided insights on how students were reacting to certain subjects. The students in my Honors class on "Understanding Scientific Reasoning" were most helpful by providing feedback and discussing how they actually study (or don't).

References

1. Turing, A.: Computing machinery and intelligence. Mind LIX (Oct. 1950) 433–460 doi:10.1093/mind/LIX.236.433.
2. Stevenson, D.E.: The problem with problems in computational science and engineering problem-based learning. In: 2006 International Conference on Computational Science and Education. (Aug 2006)
3. Bushey, D.E.: Critical thinking traits of top-tier experts and implications for computer science education. PhD thesis, Clemson University, Clemson, SC (August 2007)
4. Parham, J., Chinn, D., Stevenson, D.E.: Using bloom's taxonomy to code verbal protocols of students solving a data structure problem. In: Proceedings of ACM SE 2009. (2009) Submitted.
5. Walton, D.: Informal Logic: A Pragmatic Approach. Cambridge University Press, Cambridge (2008)
6. Johnson-Laird, P.: Mental Models. Harvard University Press (1983)
7. Craik, K.: The Nature of Explanation. Cambridge University Press, Cambridge, England (1943)
8. Davidson, J.E., Sternberg, R.J.: The Psychology of Problem Solving. Cambridge University Press, Cambridge, UK (2003)
9. Anderson, J.: Language, Memory, and Thought. Erlbaum (1976)
10. Marshall, S.P.: Schemas in Problem Solving. Cambridge University Press (1995)
11. Flavell, J.H.: Metacognitive aspects of problem solving. In Resnick, L.B., ed.: The Nature of Intelligence. Erlbaum, Hillsdale, NJ (1976) 231–236
12. Johnson-Laird, P.: How We Reason. Oxford University Press (2009)
13. Stenning, K., van Lambalgen, M.: Human Reasoning and Cognitive Science. Bradford Books (2008)
14. Walton, D.: Fundamentals of Critical Argumentation (Critical Reasoning and Argumentation. Cambridge University Press (2005)
15. Freeley, A.J., Steinberg, D.L.: Argumentation and Debate: Critical Thinking for Reasoned Decision Making. 10th edn. Wadsworth (1999)
16. Zarefsky, D.: Argumentation: The Study of Effective Reasoning. The Thinking Company (2005)
17. Giere, R.N., Bickle, J., Mauldin, R.: Understanding Scientific Reasoning. Wadsworth (2005)
18. Mattson, T.G., Sanders, B.A., Massingill, B.L.: Patterns for Parallel Programming. Addison-Wesley (2005)

A Mathematical Modeling Module with System Engineering Approach for Teaching Undergraduate Students to Conquer Complexity

Hong Liu and Jayathi Raghavan

Embry-Riddle Aeronautical University
600, S. Clyde Morris Blvd, Daytona Beach, FL, 32114
liuho@erau.edu, raghavaj@erau.edu

Abstract. This paper presents a mathematical modeling module for ODE courses. The module uses light-weight systems engineering approach to promote the competency of undergraduates to overcome the complexity in applied mathematics problems. The mathematics training of undergraduates in most colleges is limited to solving applications with a couple of variables in few steps of computations. Once faced with problems beyond that level of complexity, they are not only challenged to plan a scheme for finding solutions, but also to provide justification for their answers. This module combines an iterative modeling process with the compartmental analysis methodology to leverage these challenges. Verification and validation techniques are introduced for assuring the soundness of answers. The query-based process forces the students to trace critical mathematics equations to the corresponding phenomena of the problem under consideration. Examples within the module are arranged with incremental complexity. Stella is used as a modeling and simulation tool

Keywords: Compartmental Analysis, Validation and Verification, Query-Based Modeling Process, Kolb Cognitive Complexity.

1 Introduction

The mathematics training of undergraduate students in most colleges is too simplistic and does not train students to deal with more complex applications. Simplistic problems usually involve limited number of variables with very few steps of computations. Once faced with problems beyond that level of complexity, they are challenged to plan their scheme to solve it. Even if they manage to find the solution, most students are unable to justify their answer. This module combines a mathematical modeling (MM) process with the compartmental analysis methodology to deal with these challenges. The case studies in this module demonstrate how to divide a complex problem into smaller problems at a level comprehensible for our students. The models in each case study are organized with increasing complexity so that students can incrementally progress from simple to more complex applications. A new model is obtained from previous ones by gradually relaxing unrealistic assumptions and considering additional factors [9] and [10]. Verification and validation (V&V) techniques

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 93–102, 2009.

[7] are introduced to ensure the validity of the answers. Instructive questions are designed to guide students to obtain their own answers and force them to trace critical mathematics equations to the corresponding phenomena of the problem under consideration. With the use of appropriate computational tools, students can obtain numerical solutions for the complicated system of equations that they cannot solve analytically [1]. We choose Stella by ISEE systems [3] and [4] as our main modeling and simulation tool. This module illustrates to students how mathematics rigor can be combined with visual intuitiveness of computational tools to obtain insightful answers to mathematical applications.

2 Outcomes and Assessment Plans

The objectives of the module are as follows: 1: Teach students basic modeling methodology and process. 2: Train students to use tools to model an application incrementally. 3: Introduce the concepts of validation and verification. 4: Cultivate students' ability to relate their answers to the insightful observations to the applications. 5: Demonstrate the power of mathematics as a scientific language. The original module outlined the outcomes for the knowledge, skills and competency (KSC) of the module, described the Kolb Complexity level defined in [11], and provided an assessment plan [2] and [6] associated with outcomes. It is omitted here due to space limitation. The summative evaluation consisted of 5 survey questions associated with each of the 5 objectives mentioned above and two questions relating to the overall feedback of the module delivery and the team projects.

3 Mathematical Modeling Process

Mathematical Modeling is an interdisciplinary subject that applies systems engineering principles to mathematical applications. It provides the methodology and the process to transform application problems to mathematical problems. A mathematical model uses mathematical language to describe a system. There are several types of mathematical models and modeling methodology. Scientists and engineers often model two types of changes: a continuous quantitative change such as change of velocity, or a discrete qualitative change such as changing traffic lights. The continuous phenomenon is typically modeled by differential equations; and the discrete phenomenon is typically modeled by state machines or graphs. We would like to focus on the differential equation models of continuous phenomena.

Every branch of systems engineering follows a process and some kind of methodology to model a system. We present an iterative modeling process as illustrated by the schematic diagram Fig. 1. It is a recommended general framework for working through a problem from model formulation to answer justification. The process consists of two reverse loops. The counterclockwise loop indicated by solid arrows is called Problem Solving Loop (PSL) and the clockwise loop indicated by dotted arrows is called answer justifying loop (AJL). The steps in both the loops are to be iterated until a satisfactory answer to the original problem has been reached.

PSL1: Identification of Facts, Assumptions and Invariants. What are the relevant factors; such as known or unknown quantities and their units? What are the invariants; such as physics laws, etc.? These invariants are called conceptual models and are observable from domain knowledge of the concerned application or simple common sense. It can be described in natural language such as English. What are the assumptions that the conceptual model is based on? How realistic are those assumptions ranging from most idealistic to most realistic? What are the tradeoff decisions between the fidelity of the model and the easiness to find solutions?

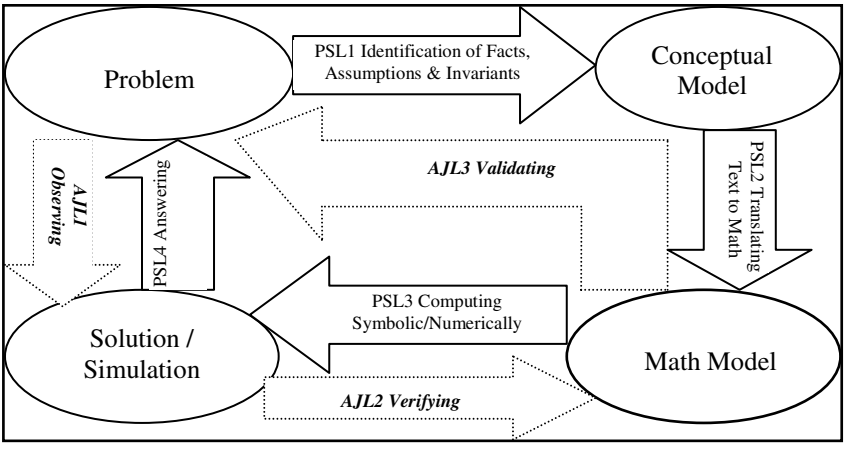


Fig. 1. The counterclockwise loop PSL and clockwise loop (dotted arrows) AJL

PSL2, Translating the Identifications above to Mathematical Expressions. Specify a coordinate or reference frame system to indicate each known or unknown quantities as parameters and variables. Use a table to map each mathematical variable, parameter to its designated quantity, positive direction, and unit. Translate the conceptual models to mathematical models using the mathematical notations specified above. The symbolic representations of the conceptual models lead to primitive mathematical models.

PSL3, Computing Symbolically or Numerically. What type of equation is it (e.g. linear or nonlinear), does it have a solution, and is the solution unique? Is it possible to obtain exact analytic solution or is only numerical solution possible? What tools and programming languages could be used to obtain the solutions?

PSL4, Answering. Does the problem require a functional relationship or a particular numerical answer? A functional relationship may be answered by an analytic function, a graph, or a simulation by a computational tool.

AJL1, Observation. What are the observable facts that are related to the answers in special points such as initial, average, median, or terminal? What is the long term trend? Can the answers be used to explain the phenomenon in terms of the domain

knowledge of the application problem? In most scientific or engineering fields, it depends on available data.

AJL2, Verification. Verification is the action to check whether the analytical solutions are correct or whether the numerical solutions approximate the exact ones within a controlled range of error. Is there any analytic way to check the answers? If numerical solution is given, what is the relative accuracy? Is it possible to obtain analytic solution in some special cases to check the numerical solution? For example, is it possible to use phase plane equation to check the ratio of two related variables of numerical solution? Is the algorithm correct and how can it be checked? Is the sign or magnitude correct and how can they be checked?

AJL3, Validation. Validation is the task of checking if the assumptions of the model are based on sound domain knowledge of the problem and if the essences of problem are properly formulated by the model. Do the resulting equations correctly model the application problem? How can the assumptions be justified? Are there any assumptions highly unrealistic and can they be relaxed so as to improve the fidelity of the model? Do the units on both sides of the equations match? Is it possible to get an intuitive understanding of some qualitative answers? If so, does the model make sense by simply observing the qualitative relationships of variables? What are the special case scenarios of your model so that it reduces to one of the known basic models?

4 Compartment Analysis

Stella meta-models embody the compartment analysis methodology. To help students understand Stella diagrams, the original module of this lecture started with the introduction of the basic idea of Stella. Then, we discussed the conceptual model of single compartment system. Due to space limitations, this paper only presents the conceptual model of multiple compartment system and two examples, with one focusing on the compartmental methodology and the other on the process.

Most natural or artificial system is composed of components each of which fulfills certain functionality and interacts with others in certain patterns. A well organized component can be abstracted as a black box, where only the inputs from and outputs to the other components or the environment matter to the problem under consideration. We define such a component of a system as a compartment [5]. The methodology of modeling and analyzing a system as functionally decomposed compartments, according to their interacting patterns is called compartmental analysis. Since only inputs and outputs matter, each of them can be treated uniformly and systematically as follows. Let x_1, x_2,… x_n be the n unknown compartmental variables that are functions of time t, $input_{ii}$, $output_{ii}$ represent the input from and output to the environment, $input_{ji}$, $output_{ij,}$ represent the input from the j-th compartment to the i-th compartment and the output from the i-th compartment to the j-th compartment respectively that are possibly functions of time t, where $1 \le i \le n$ and $1 \le j \le n$. Then, Fig. 2. describes the quantitative exchange of the compartmental variables x_i to other compartments and environment.

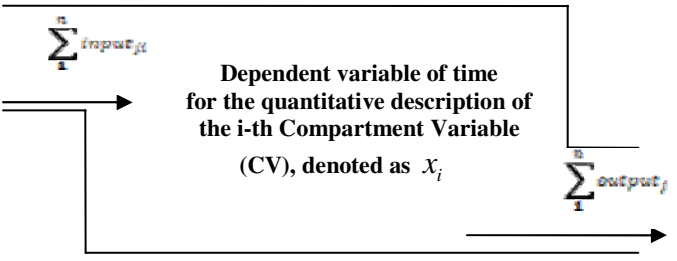


Fig. 2. Diagram for the *i*-th compartment of a system

Only the inputs and outputs contribute to the quantitative changes of the CVs. The rate of change of the *i*-th CV over unit time = The *i*-th CV now –The *i*-th CV a unit time ago. That is:

$$\frac{dx_i}{dt} = \sum_{j=1}^{n} [input_{ji}(t) - output_{ij}(t)] \text{ where } 1 \le i \le n, \tag{1}$$

Since there are *n* similar equations corresponding to each of the *n* compartments, we obtain an *n*×*n* system of equations along with the *n* initial conditions.

Example 4.1. A large tank holds 1000 L of brine solution of salt with concentration 1 Kg/L. A brine solution with low concentration of salt begins to flow into the tank at a constant rate of 4 L/min to dilute the resolution and the well stirred resolution flows out of the tank at the same rate of 4 L/min. If the solution of the brine flowing to the tank has concentration of 0.2 Kg/L, determine when the concentration of the salt of the brine will be diluted to 0.6 Kg/L. What is the concentration of the brine after very long time?

Solution. The focus of this example is the basic idea of compartmental analysis, so, we skip many process details and calculations. We find that the variables are the independent variable *t*: time and the compartment variable *x*(*t*): the amount of pure salt at time *t*. We observe that the volume *V* of the brine in the tank is invariant since the rate at which the solution flows in is equal to the rate at which the solution flows out. We list the assumptions as follows: 1: The flow continues indefinitely, 2: The brine inside the tank has uniform concentration , 3: The only change in the amount of pure salt in the compartment is due to the difference between the amount of salt flowing in (input) and out (output).

Now, we consider the change of *x*(*t*) per unit time on both sides of the conceptual model in (1) 1 at *t*, we have: *input(t) = 0.8 Kg/Min*, *output(t) = x(t)/250 Kg/Min.* Initially, *x(0) = 1000 Kg*. By (1), we have the initial value problem and its solution as

$$\begin{cases} x' = 0.8 - x/250 \\ x(0) = 1000 \end{cases} \tag{2}$$

$$x(t) = 200 + 800e^{-t/250} \tag{3}$$

Answers. The concentration of the salt of the brine will be diluted to 0.6 Kg/L after 152.25 minutes, and will be close to 0.2 Kg/L after a very long period of time.

Example 4.2. Two tanks, each holding 100 liters of a brine solution, are interconnected by pipes. Fresh water flows into tanks A at a rate of 6 L/min and the fluid is drained from tank B at the same rate. If 8 L/min of fluid are pumped from tank A to tank B, and 2 L/min from tank B to tank A, and the liquids insides each tank are well stirred so that the mixture is homogeneous, also assume that the brine in tank A contains 6 kg of salt and brine in tank B contains 60 kg of salt initially, determine the mass of salt in each tank at time t = 10 minutes, 15 minutes and 24 minutes.

Stella incorporates the compartment analysis methodology in its meta-models. The analogy of interconnected flows and reservoirs between Stella components, allows users considering a system as compartments, exchange quantities through Stella's input and output interfaces. We adapted a few steps of the process to demonstrate how to use the process and Stella to solve the problem.

PSL 1. Facts a: Variables: Amount of pure salt in tank A and tank B - APureSalt, and BPureSalt, b: All other facts identified will be represented by converters in the Fig. 3. Assumptions a: The flow continues indefinitely, b: The brine inside the tank has uniform concentration, c: The change of salt results from the difference between input and output. Invariants a: The change rate of APureSalt per min = AFlowIn + FlowBtoA – FlowAtoB, b: The change rate of BPureSalt per min = FlowAtoB – FlowBtoA – BFlowOut. A conceptual model is shown in Fig. 3.

PSL 4, **Answer the question.** The solutions are given by either a graph or a table. The amount of pure salt in tanks A and B at times t =10, 15 and 24 are: (8.86, 33.53), (8.32, 25.31), and (6.58, 16.11) Kg, respectively.

AJL1, Observe the answer. Observe the answers in special points such as initial, or terminal, and the long term trend. The amount of salt in both tanks can be found to be reducing simultaneously, which confirms our understanding, since pure water flows in to dilute the brine solutions which were in the tank.

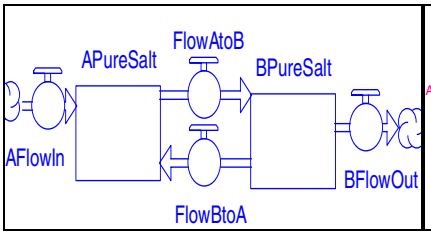


Fig. 3. Conceptual Model

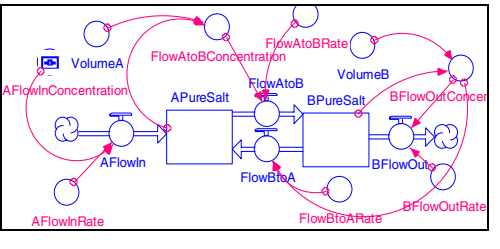


Fig. 4. Mathematical Model

5 The Motion of Floating Objects

In the original module, we gave two examples to study the oscillation of spring-mass system, one is a simple spring-mass system and the other is a coupled spring-mass

system. Students learned how to introduce intermediate variable to transform higher order equations to system of first order equations and then to solve them using Stella. Here, we will study the motion of floating objects based on the understanding of the oscillation of spring-mass systems.

Consider the motion of a floating object with mass m, as illustrated by the Fig. 5. Let z be the displacement of the centroid of the object from the surface of the water, $V(z)$ represent the volume of the object submerged, ρ the density of water, b the damping coefficient of water friction. Then, the weight $W = -mg$, (g is the acceleration due to gravity) the buoyant force $B = \rho g V(z)$, frictional force $f = -b\,z'(t)$, where $z'(t)$ is the velocity of the floating object. By Newton's second law, we have $a_z = B + W + f$ and $a_z = z''(t)$, hence

$$mz''(t) = \rho g V(z) - mg - bz'(t) \tag{4}$$

Rewriting it in standard form for second order ODE, we get:

$$z''(t) + (b/m)z'(t) - (\rho g/m)V(z) + g = 0 \tag{5}$$

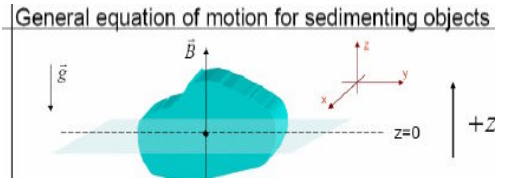


Fig. 5.

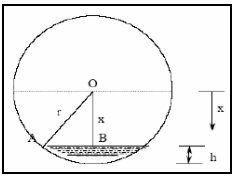


Fig. 6.

Example 5.1. A solid cube that has 2 feet side length is initially released on the water surface from rest. If the density of the solid is a half of the water's density ρ and (a) the water friction can be neglected, (b) the damping coefficients of water friction is 499.2 lb-sec/ft, find the equation that governs the oscillation of the cube. (c) Build a general model for the motion of a floated solid cube with its density below that of water and its friction coefficient flexible from zero to 1000 lb-sec/ft, and then use the parameters given in (a) and in (b) to simulate the motion of the cube.

Solution. Let z be the displacement of the centroid of the object from the surface of the water, $V(z)$ the volume of the object submerged, L the length of the cube, then

$$V(z) = L^2(L/2 - z) = L^3/2 - L^2 z \tag{6}$$

Because the equilibrium point of the cube is where the buoyant force equals the mass of the object, we have: $\rho g V(z) = mg$, $m = L^3(\rho/2)$, since the density of the cube is $\rho/2$ where ρ is the density of water, then $m = L^3(\rho/2)$, and instantiating all parameters with known constants, we get (7) for case (a) and (8) for case (b).

$$\begin{aligned} &z''(t) + 32\,z = 0 \\ &z(0) = 1; z'(0) = 0 \end{aligned} \tag{7}$$

$$z''(t) + 2z'(t) + 32z = 0$$
$$z(0) = 1, z'(0) = 0 \tag{8}$$

Notice that (7) and (8) are exactly the same as the equations for spring mass system. Hence, we observe that the motion of a cube in water is similar to the oscillation of a spring mass system.

(C) If the density of the cube is an arbitrary constant δ, less than that of water, then, we do not get z = 0 at the equilibrium point. Depending on whether the density is more or less than half of the water's density, z at equilibrium point will be negative or positive, respectively to balance the weight. The mass of the cube is $m = L^3\delta$, substituting (6) in (5), we have

$$z''(t) + b/(\delta L^3)z'(t) + (\rho g/(\delta L))z + (1 - \rho/(2\delta))g = 0 \tag{9}$$

Equation 9 is a non-homogeneous second linear equation. It can be easily transformed into a homogeneous equation in y. Let z = c be the equilibrium point, we have $\rho g/(\delta L)c + (1-\rho/(2\delta))g = 0$, Solving for c, we get $c = L(\rho - 2\delta)/(2\rho)$, Substituting $y = z - c$, we get $y = 0$ as the equilibrium point. Equation (9) now becomes

$$y''(t) + b/(\delta L^3)y'(t) + \rho g/(\delta L)y = 0 \tag{10}$$

If the initial conditions are applied and if the water friction is small enough, the solution is almost the same as that of (8), except the attenuating oscillation is around the new equilibrium point $z = L(\rho - 2\delta)/(2\rho)$, instead of z = 0. The Simulation is illustrated in Fig. 7 when $\delta = 20.8 = 1/3\rho$ and friction coefficient b=115.

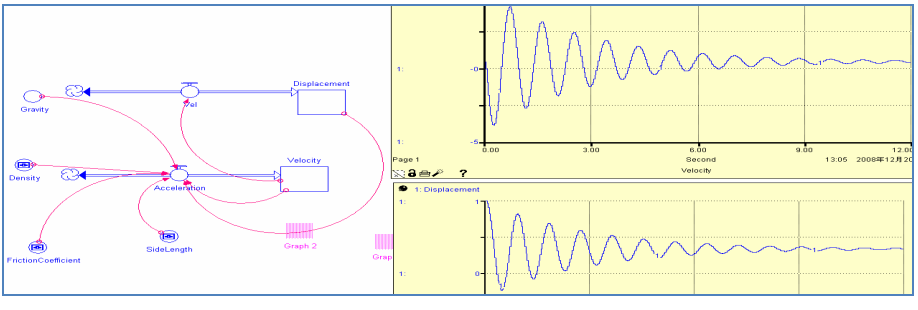


Fig. 7. The Stalla Model and Simulation of the Velocity and Displacement

Our next example in this section will discuss the oscillation of floating sphere. Our focus is to find how to calculate the $V(z)$ in (5). We would like to develop a general model for a solid sphere of radius r and density δ that is less than the water's densityρ. The volume $V(h)$ of the submerged spherical cap illustrated in Fig. 5.2, where h is the height of the sphere submerged in water. Let the distance from the center of the sphere to water level be x. Consider the right triangle OAB, by Pythagoras

Theorem $AB=\sqrt{r^2-x^2}$. The section area of part of the sphere is $A(x) = \pi(AB)^2 = \pi(r^2 - x^2)$, $V(h) = \int_{r-h}^{r} A(x)dx = \int_{r-h}^{r} \pi(r^2 - x^2)dx = \frac{\pi h^2(3r-h)}{3}$.

As shown in Fig. 6, let z = 0 be the water surface level, and $z = x = r - h$ be the displacement of the center of the sphere from the surface of the water, we have $h = r - z$ and $V(z) = \pi(r - z)^2(2r + z)/3$. By (5), we have the differential equation for the motion of the floating sphere as follows

$$z''(t) + \frac{3b}{4\pi r^3\delta}z'(t) - \frac{\rho g}{4r^3\delta}(z^3 - 3r^2z) + (1 - \frac{\rho}{2\delta})g = 0 \tag{11}$$

Example 5.2. Change the solid cube in example 5.1 to a solid sphere of 1 foot radius, build the mathematical models and solve them by using Stella for all three cases.

Solution. By assumptions, substitute the parameters in (11) with the constants from the problem description, we have (12) for case (a) and (13) for case (b)

$$\begin{cases} z''(t) + 48\,z(t) - 16\,z^3(t) = 0 \\ z(0) = 1; z'(0) = 0 \end{cases} \tag{12}$$

$$\begin{cases} z''(t) + 3.82\,z'(t) + 48\,z(t) - 16\,z^3(t) = 0 \\ z(0) = 1, z'(0) = 0 \end{cases} \tag{13}$$

The Stella model of case (c) is similar to case (b) except for the parameters in (11). The solutions for case (a) and (b) are illustrated in Fig. 8 and Fig. 9. respectively.

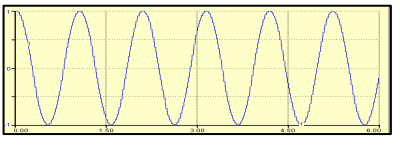


Fig. 8.

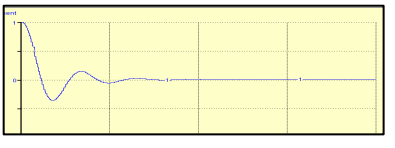


Fig. 9.

Validation. A critical assumption is that the motion of the object is constrained to a half of the side length or the radius so that neither does the object jump out of the water nor is the whole body of the object submerged below water surface. A question to help students understand the assumption is: If we change our initial conditions, such as adding an initial downward velocity besides the given initial displacement, are our models in this section still valid? If not, why?

6 Conclusion

This paper presents an abbreviated version of a mathematical modeling module, which is selected and evolved from the course material that the authors have been delivering in their ODE courses and the seminars of SIAM student chapter [8]. Section 1 presents the motivation of the module. Section 2 lists the objectives of the module and assessment plan. Section 3 describes an iterative modeling process. Section 4 illustrates the

compartmental analysis methodology. The examples in section 4 and 5 cover a series of models in orders of incremental complexity spanning from first order linear equation to second order nonlinear system of equations. The examples are connected as they gradually increase in difficulty and complexity. This connection of the examples allows the students to comprehend the material in small doses and minimizes the intimidation to the students by the complexity of the problems and motivates them to confidently work on team projects with similar level of complexity.

Acknowledgment. The authors would like to give thanks to the Summer Workshops by Shodor for introducing us to computational science education research.

References

1. National Computational Science Institute, Incorporating Computational Science Tools and Techniques into Undergraduate Course, Workshop for Undergraduate Faculty (2004)
2. Mathematical Science Education Board, Measuring What Counts: A Conceptual Guide for Mathematics Assessment. National Academy Press, Washington (1993)
3. Richmond, B.: An Introduction to System Thinking, Stella Software. High Performance System Inc.
4. Stella, High Performance System Inc., http://www.hps-inc.com
5. Nagel, Saff: Fundamentals of Differential Equations, 6th edn. Addison Wesley, Reading (2004)
6. Falmagne, J.-C.: The Assessment of Knowledge in Theory and in Practice, http://www.aleks.com/about_aleks/research_behind
7. Liu, H., Gluch, D.P.: A Proposal for Introduction Model Checking into an Undergraduate Software Engineering Curriculum. The Journal of Computing Science in Colleges 18(2), 259–270 (2002)
8. Liu, H.: Offering Honors Course Option within an Ordinary Mathematical Course for Undergraduate Students in Engineering Majors. In: The Proceedings of the 2008 ASEE Annual Conference (2008) (to appear)
9. Vakalis, I., Karkowski, A., Lahm, T.: A Guidebook for the Creation of Computational Sci. Modules, http://oldsite.capital.edu/acad/as/csac/Keck/guidebook.html
10. Berkey, D.: A Model for Vertical Integration of Real-world Problems in Mathematics. In: The Proceedings of ASEE Annual Conference & Exposition (2007)
11. Richlin, L.: Blueprint for Learning: Creating College Courses to Facilitate, Assess, and Document Learning. Stylus Publishing (2006)

Lessons Learned from a Structured Undergraduate Mentorship Program in Computational Mathematics at George Mason University

John Wallin[1] and Tim Sauer[2]

[1] Department of Computational and Data Sciences, College of Science, George Mason University, Fairfax, VA 22030, USA
[2] Department of Mathematical Sciences, College of Science, George Mason University, Fairfax, VA 22030, USA

Abstract. We present the results from the first two years of the Undergraduate Research for Computational Mathematics (URCM) program at George Mason University (Mason). In this program, students work on a year-long research project in Computational Science while being supervised by a mentor. We describe the structure and goals of this program along with some observations about the elements that we have found that have been challenging in its implementation. Finally, we will provide a summary of the outcomes of the first two years of this project.[1]

1 Introduction

In the Spring of 2007, the Department of Mathematical Sciences and the Department of Computational and Data Sciences (CDS) at George Mason University received funding from the National Science Foundation to create an undergraduate mentorship program in computational mathematics. This project was funded through the Computational Science for Undergraduates in the Mathematical Sciences (CSUMS) program for a five one-year cohorts. In the spring of 2007, the project selected the first ten students in this year-long program. A second cohort was selected in the spring of 2008.

NSF originated the CSUMS program for students in the mathematical sciences "to better prepare these students to pursue careers and graduate study in fields that require integrated strengths in computation and the mathematical sciences."[1] Unlike the NSF funded Research Experiences for Undergraduates (REU) program where students work on a project over the summer, the CSUMS program is allows students to work on their own independent research projects over a full year under the direction of a faculty mentor. For the students in our

[1] The development of the GMU Computational and Data Sciences undergraduate program is sponsored by the NSF CSUMS (Computational Science Training for Undergraduates in the Mathematical Sciences) program, through award # DMS-0639300.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 103–110, 2009.

program, this was their first exposure to working on a long term research project. As summarized by the National Academy of Sciences, ". . . the one-on-one mentoring that takes place in supervised undergraduate research is one of the best predictors of students' professional success." [2]

The goal of the George Mason University URCM program is to create a one-on-one mentoring program in the computational sciences for undergraduate students in the mathematical sciences. In this program, we hope to enhance the professional success of our students by placing them in supervised, independent research projects. This is being done by

- exposing undergraduate students to the applied mathematics, science, and computer science used in cutting-edge computational science projects through mentorship, classes, and seminars,
- developing the formal and informal learning environments that allow students to be successful at developing, completing, and presenting their projects,
- developing a peer learning community by encouraging structured student interaction through classes, seminars, labs, as well as formal and informal meetings in their research group,
- creating close ties between student and mentor, where the student can receive ongoing feedback on the project's progress as well as constructive feedback on personal strengths and weaknesses in becoming a professional mathematician and scientist,
- extending this program beyond the five-year period that is funded by NSF by matching undergraduate students to scientists who have existing grants both on the Mason campus and in the larger community.

The students in the program work in a year-long funded research project under the direction of a faculty mentor. Because of NSF guidelines, all the students in the program are required to be mathematics majors. In our program, these students are generally in their junior or senior year. Students apply in March, and a selection of the students for the yearly cohort is made by the beginning of April.

The students who have participated in this program have worked on a wide variety of computational science research. Although individual research projects ranged from chemical phase transitions and material structure to computational hemodynamics, they all had the common theme of numerical and mathematical modeling of physical systems. Unlike with individual student research programs, this common theme provides a unifying element of the program that is used to create support structures that benefit all the students.

Student participation in the project begins at the beginning of the first summer session in late May. During this summer semester, all the students are required to take a course in Numerical Partial Differential Equations (PDE). This class is designed to fill in the background the students need for their research projects as well as help build the peer relationships and support structure they will need as they transition to research. As a prerequisite, we ask the our student complete a one-semester course in basic numerical methods. In parallel with the numerical PDE course, the students also participate in a daily lab session where they learn how to use basic computational tools and programming languages. After the PDE

class ends in early July, the students continue to attend the lab session twice a week and start working with their advisors on research projects. Throughout the summer, these lab sessions are kept very informal and are designed to help fill in the computational background they need for their research projects.

In both the Fall and Spring semester, students are required to attend a one-credit seminar that meets once a week. In this seminar, students are given information on careers, publications, graduate school, and also required to present frequent updates on their research. A major focus of the seminar is to let students polish their public speaking and writing skills as they prepare to go to conferences, create their posters about their work, and write papers. The seminar also provides a regular venue for students to discuss the problems they are having in their research projects.

In the URCM program at Mason, two students are paired together with one or more faculty mentors. The students work on different aspects of the same project, so they can have a peer support network outside the classroom. The students may sign up up to six credits for undergraduate directed reading and research over their research project so they may receive academic credit for their work. The students are also given a $10,000 stipend over the year long project, and loaned a laptop computer so they have a portable platform for their computing and presentations. Travel to conferences, software costs, and publication charges are also covered in the grant.

2 Observations

As the Director of the mentorship program, Dr. Wallin worked with all the students in the summer laboratory and then interacted with them the weekly seminar that he directed. He also interacted with all the mentors on a weekly basis. Dr. Sauer was the PI for the grant, and also worked closely with both the fauclty and the students. In these capacities, we have compiled feedback from both the students and the mentors. Because only thirteen students have been in this program so far, the results are anecdotal. We will be using the pronoun "we" to describe these findings, since they reflect the informal consensus of the groups involved. We believe that these observations reflect some of the underlying issues that others might encounter in similar undergraduate research programs in computational science.

First, the success of students seemed to be somewhat correlated on the complexity of the software that they used in their projects. If students developed their own simple codes, it seemed they remained more engaged and motivated in the research work. This finding may be more directly related to the students being mathematics majors, not computer science or computational science majors.

Before starting work on their projects, our students had not been exposed to the challenges of working with "black box" programs. We define the term "black box" in this context to mean programs too complicated for any one person (particularly an undergraduate mathematics student) to completely understand. These types of programs are commonly used in complex calculations across the

computational sciences to simulate diverse problems such as fluid flow or material science. Most of the students in our program were interested in approaching the higher level more theoretical questions rather than doing these production calculations, even if those calculations were tied to very engaging projects.

One issue with using "black box" in undergraduate research projects might be the learning curve associated with these complex systems. However, it seemed like most of the frustration that these students felt was the lack of connection between their classroom studies and their research work. While computational science projects tend to focus on projects of very high complexity with difficult interconnections, this paradigm does not seem to map as well to the typical background of undergraduate mathematics majors we found in our programs. In general, it seemed difficult to move our students beyond the simpler Matlab/Octave tools toward complex programs and compiled programming languages.

Second, the students in the program often did not engage fully in the research work until they were forced to present their work in our weekly seminar. At the beginning of the fall semester, about half of the students had not made significant progress on their research projects during the first year of the program. When they were required to make class presentations and given deadlines in their schedule, the students systematically started making good progress. The quality of the presentations, their understanding of the problem's context, and their research results all came into focus during the fall semester.

Some of the delay in progress might have been from not understanding the expectations being set by their advisors. We believe that part of the problem can be attributed to the difficulties in the transition between classroom learning and research, as well as to procrastination that all of us feel until a deadline is upon us. Even so, setting and enforcing milestones with definable metrics in these research projects seems to be very important in promoting student learning, particularly in the first months of a long-term project. Even though the students were working on independent projects, they needed regular direction from their mentors to continue making progress. As students became more vested in the research work, most began to develop their own project schedules and started integrating research more into their weekly schedule.

Third, we found that the peer support planned in this project was only partially successful at helping students overcome difficulties they encountered in their research. Often a student would turn to his research partner for help, but busy schedules and geographic considerations made it difficult to get and give help. Most of the students in our program and across Mason in general lived off campus. The commute times and lack of face to face contact outside the normal university hours probably worked against this collaboration. The students would often turn to their research advisor instead of their research peer for help.

In contrast, we did find that the students immediately started using text messaging, Facebook, and video conferencing through their laptops during their class to talk about homework assignments during the summer PDE class. This transition took place in days, perhaps even hours, after we gave them all laptops. When the students were working as a class on the same problems, they tended to

function together extremely well. This collaboration did not translate as well as we hoped when they started working on independent projects. Even though two or three students were assigned to very similar projects with the same advisor, the differences between the projects and lack of common focus seemed to be a barrier for effective peer support. In the second cohort, we worked to instill the expectations that the students in peer groups should meet on a weekly basis. This seems to have created a better system of peer support during the second year of the project.

Fourth, students who signed up for academic credit for their research work were generally more productive than those who did not sign up for these research credits. Some students who tried to balance a full class load with no time explicitly budgeted for their research activities. In the most extreme case, some students tried to sign up for a full course load during summer. When they did this, their research often was viewed as secondary activity that was pushed aside when there were no nearby deadlines. When students explicitly budgeted their time through credit hours, this problem was significantly reduced.

Fifth, peer support of mentors is just as important as peer support of students. Even experienced faculty who had supervised graduate students had some trouble effectively mentoring undergraduate students. Regular, informal meetings between the mentors seemed to help them see how other students and mentors were working together. Eventually, this helped make the mentorship experience more consistent between the students.

In our first year, the faculty mentors were split in two different buildings. This inadvertently led to less frequent contact between the faculty, despite efforts to encourage communications. During the second cohort, all the faculty mentors had offices close together leading to more regular discussions about their students and the program. It seems this extra support seemed to help the students in the program become more successful in their research projects earlier in their year-long projects.

Finally, we have found that recruiting students into this project was more difficult than expected. Even with the generous stipend, the laptop, and the experience that students would gain through the project, many students in the mathematical sciences were reticent to even apply to the program. When we asked students who were in the program why other students were not applying, we found several surprising answers. Some students in our program reported that their friends did not want to get involved in research, since they had no plans to go to graduate school. Others students reported that their friends were not interested in either doing research in or getting a job in applied mathematics and computing.

It seems that many undergraduate students in the mathematical sciences (and probably beyond) don't have a good understanding of the jobs they are likely to do after they graduate since most studies indicate that the vast majority of jobs for majors across the sciences and mathematics are IT related. [3].

3 Program Outcomes

Beyond our observations of the elements that seem to help and hinder our mentorship program, it is important to consider the tangible outcomes of this project. All but one of the thirteen students have presented both oral and poster papers for external conferences. Most of the students presented papers at both regional and national conferences, and one was given an award for his poster at the San Diego meeting of the American Mathematical Society in 2007. The confidence that the students had in creating and giving presentations along with the quality of these presentations was dramatically enhanced by the practice in the seminar class.

The skills that students exhibited with software tools also increased dramatically through the project. Most of our students had no familiarity with Unix at the start of the project, and were experts by the end of the year. The students'skills at using high level tools like Matlab/Octave also greatly increased during the program. By the end of the year, all of the students felt comfortable putting complex PDE system into Matlab or Octave. Many of these students had done very little programming at the beginning of their research project, but were very comfortable writing these types of codes by the end of their year.

The quality of the final research projects we saw in the program was very high. Most of the mentors felt that the work being done by their undergraduate students was at the level of first and second year graduate students. The students also noted this when they compared their work to other peer groups at conferences. In particular, one student reported that he felt his work was "just average" when he looked at the other posters at a conference. He didn't realize at the time that there were only two undergraduate students at this poster session of about forty graduate students.

One of the least quantifiable elements of these projects was the excitement that our students felt about being able to participate and carry out significant, independent research projects under the direction of a real scientist mentor. Having the students participate in regional and national conferences also opened up the world of mathematical and computational research to them, and let them understand the research process from conception to presentation. Overall, our students reported very positive feelings about the program, despite some of the difficulties they encountered in their projects.

Of the nine students in the first year cohort, all but one have gone on to take graduate courses after graduation. Six are pursing doctoral degrees, and two others are working masters degrees. Four of the six students pursuing the Ph.D. are studying either Computational Science or Applied Mathematics. The remaining doctoral students are working in operations research or medicine. The only student not currently enrolled in graduate studies is taking a year off before starting medical school. It is important to note that less than half of students had seriously considered graduate school before they started the project. Based on the status of the first cohort, we strongly feel that this program helped change the attitudes our students had toward research in Computational Science and Mathematics.

4 Summary and Conclusions

We have found a great deal of unexpected challenges in this project, and are still learning how to improve the experience for our mentors and our students. Based on our experiences, we make the following recommendations for other similar programs:

- Pick projects that help the students apply their classroom work directly to their research projects. Having students switch too quickly into complex software systems can make it difficult for them to make this connections, and make it seem like their studies are unrelated to their research work.
- Make sure to create and enforce milestones through out the program. Although this is an independent research project, students learning how to do research need structure particularly in the first few months of the project.
- Set expectations for peer meetings and peer support during the project. Make sure the students understand that they are expected to work together, and keep the projects in peer groups similar enough so that they can benefit from each other's experience.
- Strongly encourage students sign up for directed reading and research credits when they are in a mentorship or research program. Students tend to over schedule themselves if time isn't put into their academic schedules.
- Build informal activities to provide peer support of the mentors. This type of support is very useful in normalizing expectations and keeping the mentors engaged in the student projects.
- Try to minimize geographic barriers to collaboration both between student and between mentors. Make sure regular meetings occur within and between the different groups.

Despite some of the initial problems, we are very happy with the outcomes of our students both in terms of their research work and their careers. We feel that our program contributed to their professional development, and has provided them with a strong basis for further academic work and for their future jobs.

Now that the new major and minor in the Computational and Data Science is in place at Mason, we hope to see future double majors in this program and minors from the department of Mathematics in these programs. We feel that the minor will be particularly attractive to these students since many of the requirements are already met by their participation in this program via the numerical methods and research courses they are already taking in their mentorship. With these programs in place, we feel our students will students develop the computational skills they need in their careers or for future graduate work.

Creating projects that for the students that were independent, exciting, and cutting-edge yet suited to the technical backgrounds of our undergraduates took a high level of commitment from our mentors. Despite the challenges, we have all seen a great return on the time we invested in this project as we as we watched the first cohort progress though the program, present papers, and graduate. As

the second cohort progresses through their program, it is gratifying to see them transitioning to research faster and broadening their knowledge of how to solve cutting edge problems using computational science.

References

1. Computational Science Training for Undergraduates in the Mathematical Sciences (CSUMS), National Science Foundation Solicitation 06-559 (2008), http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=13655
2. Fox, M.A., Hackerman, N. (eds.): Evaluating and Improving Undergraduate Teaching in Science, Technology, Engineering and Mathematics. The National Academy Press, Washington (2003)
3. Testimony before the House Subcommittee on Science, Bruce Mehlman, Technology, and Standards Subcommittee, June 24(2002), gop.science.house.gov/hearings/ets02/jun24/mehlman.htm

First Principle Study of the Anti– and Syn–Conformers of Thiophene–2–Carbonyl Fluoride and Selenophene–2–Carbonyl Fluoride in the Gas and Solution Phases

Hassan H. Abdallah[1] and Ponnadurai Ramasami[2]

[1] School of Chemical Sciences, Universiti Sains Malaysia, 11800 Penang, Malaysia
hwchems@usm.my
[2] Department of Chemistry, University of Mauritius, Réduit, Republic of Mauritius
p.ramasami@uom.ac.mu

Abstract. The anti- and syn-conformers of thiophene-2-carbonyl fluoride (**A**) and selenophene-2-carbonyl fluoride (**B**) have been studied in the gas phase. The transition states have also been obtained for the interconversion of the anti- and syn-conformers. The methods used are MP2 and DFT/B3LYP and the basis sets used for all atoms are 6-311++G(d,p). The optimized geometries, dipole moments, moment of inertia, energies, energy differences and rotational barriers are reported. This study has been extended to include solvent effect. Some of the vibrational frequencies of the conformers are reported with appropriate assignments. The results indicate that in the gas phase the syn conformer is more stable and the CCSD(T)//MP2 energy differences are 2.97 kJ/mol (**A**) and 3.02 kJ/mol (**B**) and barriers of rotation are 38.50 kJ/mol (**A**) and 36.89 kJ/mol (**B**). The structures and vibrational frequencies of (**A**) and (**B**) are not much affected by the solvents but the more polar conformer gets more stabilized. The major effect of the solvents is that energy difference decreases but rotational barrier increases. The peculiar characteristic of fluorine affecting conformational preference is not observed.

Keywords: Thiophene-2-carbonyl fluoride, Selenophene-2-carbonyl fluoride, MP2, DFT/B3LYP, energy difference, rotational barrier, solvent effect.

1 Introduction

2-Substituted carbonyl compounds and their analogues are known to show conformation isomerism [1] and thus exist as syn- and anti-conformers. In general, the anti conformer is more stable but the nature of substituents, in particular fluorine [2], and polarity of the medium [3], can affect the preference for a given conformer. In a previous communication [4], we reported a conformational study of the furfural, thiofurfural and selenofurfural in the gaseous and solution phases in order to understand the effect of substituting the oxygen of the carbonyl group with sulfur and selenium leaving the aromatic ring unchanged. In literature, there have been some attempts to look into the effect of changing the oxygen of the ring with sulfur and selenium. A brief survey of literature has been helpful to set the objectives of this work.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 114–121, 2009.

Braathen *et al.* [5] investigated the thiophene-2-aldehyde by microwave, infrared and Raman spectroscopy and by electron diffraction in the gaseous phase. They found that the syn-conformer is more stable than the anti-conformer. Han *et al.* [1] studied the rotational equilibria of 2-substituted furan and thiophene carbonyl derivatives using theoretical methods both in the gaseous and solution phases. They also found that the syn-conformer is always more stable and in the gaseous phase, MP2/6-31+G(d,p) level predicts that the syn form is more stable by 1.44 kcal/mol for thiophene-2-aldehyde and 0.14 kcal/mol for thiophene-2-carbonyl fluoride. Concistrè *et al.* [6] studied the structure and conformations of 2-thiophenecarboaldehyde from partially average dipolar couplings derived from proton NMR spectra. They found that the syn form of thiophene-2-aldehyde is more stable. Fleming *et al.* [7] studied the syn- and anti-conformers of thiophene-2-aldehyde using density functional method and normal coordinate analysis. They found that the results obtained theoretically are in agreement with experimental statements of the literature. Apart from these, there are increasing possibilities for the synthesis of selenophene-2-carbonyl [8] and derivatives of title compounds have wide applications in the industry [9,10].

Therefore although thiophene-2-carbonyl fluoride (**A**) has received attention, to the best of our knowledge, in literature, there is no conformational study for selenophene-2-carbonyl fluoride (**B**). We hereby report an extensive conformational study of (**A**) using higher level methods and extend our study to the selenium analogue. In this paper, the molecular structures, energy differences (ΔE) between the syn- and anti-conformers, rotational barriers, rotational thermodynamics, and vibrational spectra have been obtained for conformers of the title compounds, Fig 1, using theoretical methods. The findings of this work are hereby reported.

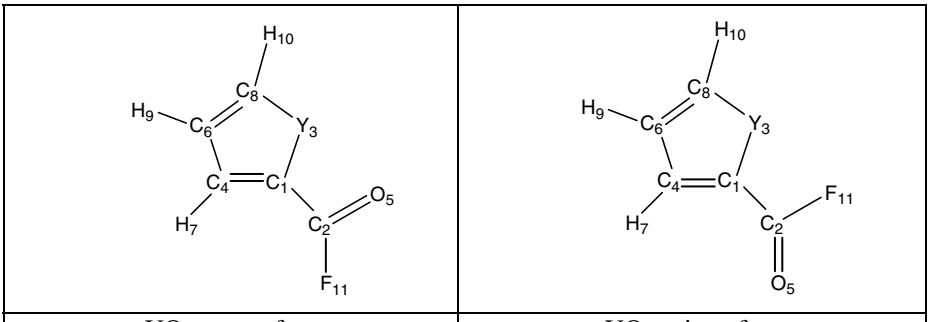


Fig. 1. Structures and atom labels of thiophene-2-carbonyl fluoride (Y=S) and selenophene-2-carbonyl fluoride (Y=Se) in C_s symmetry

2 Methods

All computations have been done using Gaussian 03W program suite [11] and Gauss View [12] has been used for visualizing the conformers.

DFT/B3LYP and MP2 methods have been used for molecular geometry optimization of the syn- and anti-conformers of (**A**) and (**B**). The basis sets used for all atoms are 6-311++G(d,p). The syn- and anti-conformers have been studied in C_s symmetry. The transition state arising from syn and anti isomerization has also been modeled. The transition state involves the planes containing $C_2O_5F_{11}$ and the ring containing hetero atom being at right angle and thus has been considered in the C_1 symmetry. Frequency computations have been carried out using the optimized structures to confirm the nature of the stationary points. Single point computations have also been carried out at the CCSD(T) level using the MP2/6-311++G(d,p) optimized structures.

Solvent effects have also been investigated by varying the dielectric constant from 1.92 (heptane) to 78.39 (water). All solution phase computations have been carried out using the integral equation formalism in the Polarizable Continuum model (IEF-PCM) [13-16] and all conformers have been fully optimized.

3 Results and Discussion

Some of the structural optimised parameters of the syn-, transition state and anti-conformers of the title compounds in the gas phase are reported in Table 1. Several conclusions can be drawn from Table 1. Firstly, there is little difference between the values of the different parameters obtained at B3LYP and MP2 level. Secondly, there is a good comparison between some of the computed parameters and those reported at HF/6-31+G(d,p) [1]. However, we believe that our structural parameters are more reliable due to the higher level of the methods used. Thirdly, the moment of inertias calculated for the conformers follow the order $I_A > I_B \approx I_C$. Fourthly, the syn conformer is more polar than the anti conformer due to opposing dipoles in the latter. Apart from these, comparing these conformers with parent molecules [4,17], it is found that hydrogen for fluorine substitution does not lead to much changes in the different optimized structural parameters except those where sulfur and selenium are involved. Further, it is interesting to note that the C1-C2 bond length is comparable in the syn- and trans-conformers but the same bond, about which rotation occurs for syn-anti isomerization, is longer in the transition state. However the other optimized structural parameters are almost comparable in the conformers for a given carbonyl fluoride.

Table 2 summarizes energies of the anti-, transition state and syn-conformers of the title compounds in the gas phase. Further, the energy differences between the anti- and syn-conformers (ΔE), the barriers of rotation between anti conformer and transition state (B1) and some thermodynamic functions (ΔH and ΔG) are also reported. The results from Table 2 indicate that (i) the syn conformers are more stable; (ii) the rotational barriers are larger than the energy differences; (iii) the energy differences and barriers of rotation are comparable for the three methods used; (iv) the effect of substituting hydrogen of the –CHO group by fluorine does not affect conformational preference but the rotational barriers and energy differences decrease; (v) the negative free energy changes indicates that the equilibrium favors the syn conformer and this is further reflected by the equilibrium being more populated with the syn conformer. The stability of the syn conformer of (**A**) is in agreement with the work of Han *et al.* [1] and they reported the energy difference as –0.59 kJ/mol computed at the MP2/6-31+G(d,p) level.

Table 1. Some optimized parameters of the syn-, transition state and anti-conformers of thiophene-2-carbonyl fluoride and selenophene-2-carbonyl fluoride computed in the gas phase

	OY-syn conformer		OY-transition state		OY-anti conformer	
	Y=S	Y=Se	Y=S	Y=Se	Y=S	Y=Se
B3LYP/6-311++G(d,p)						
Bond length (Å)						
r(C_1-C_2)	1.453 (1.462)	1.451	1.482 (1.464)	1.479	1.454	1.453
r(C_1-Y_3)	1.744 (1.730)	1.886	1.739 (1.731)	1.886	1.744	1.887
r(C_1-C_4)	1.378 (1.356)	1.374	1.368 (1.356)	1.364	1.378	1.374
r(C_2-O_5)	1.188 (1.172)	1.190	1.182 (1.171)	1.183	1.188	1.188
r(C_2-F_{11})	1.372 (1.326)	1.373	1.367 (1.327)	1.368	1.374	1.377
Angle (°)						
∠($C_2C_1Y_3$)	119.8 (111.7)	119.9	121.3 (127.3)	121.7	123.4	123.7
∠ ($C_2C_1C_4$)	128.7	128.5	127.0	126.7	125.3	124.9
∠ ($Y_3C_1C_4$)	111.5	111.6	111.7	111.6	111.4	111.4
∠ ($C_1C_2O_5$)	128.9 (127.6)	128.8	128.7 (127.3)	128.8	128.4	128.8
∠ ($C_1C_2F_{11}$)	111.0 (111.7)	111.3	111.3 (112.0)	111.4	111.4	111.3
∠ ($O_5C_2F_{11}$)	120.2	120.0	120.0	119.9	120.2	119.9
∠ ($C_1Y_3C_8$)	90.8 (90.7)	86.4	91.1 (90.7)	86.6	90.9	86.5
Rotational constant (GHz)						
A	3.664	2.646	3.687	2.595	3.688	2.627
B	1.394	1.177	1.240	1.086	1.412	1.225
C	1.010	0.815	1.102	0.880	1.021	0.836
Dipole moment (Debye)						
	4.594	4.535	3.600	3.648	4.487	4.491
MP2/6-311++G(d,p)						
Bond length (Å)						
r(C_1-C_2)	1.456	1.461	1.481	1.480	1.462	1.463
r(C_1-Y_3)	1.722	1.861	1.715	1.862	1.718	1.862
r(C_1-C_4)	1.385	1.388	1.385	1.382	1.391	1.388
r(C_2-O_5)	1.204	1.195	1.191	1.191	1.192	1.193
r(C_2-F_{11})	1.368	1.363	1.361	1.362	1.367	1.369
Angle (°)						
∠($C_2C_1Y_3$)	119.9	119.8	121.7	122.2	123.5	123.8
∠ ($C_2C_1C_4$)	128.1	128.2	126.3	126.0	124.7	124.4
∠ ($Y_3C_1C_4$)	112.1	112.0	112.1	111.8	111.9	111.8
∠ ($C_1C_2O_5$)	128.1	128.1	128.3	128.3	128.0	128.1
∠ ($C_1C_2F_{11}$)	110.8	111.1	111.0	111.1	111.2	111.2
∠ ($O_5C_2F_{11}$)	121.1	120.8	120.7	120.6	120.9	120.7
∠ ($C_1Y_3C_8$)	91.3	87.1	91.71	87.3	91.6	87.2
Rotational constant (GHz)						
A	3.654	2.651	3.686	2.603	3.695	2.638
B	1.405	1.187	1.248	1.092	1.418	1.230
C	1.015	0.820	1.110	0.886	1.025	0.839
Dipole moment (Debye)						
	3.955	3.963	4.052	3.353	3.937	3.939

(Parameters in bracket are those computed at the HF/6-31+G(d,p) level, [1])

Some of the calculated infrared raw vibrational frequencies, their intensities, Raman activities and assignments of the syn- and anti-conformers of the title compounds are reported in Table 3. The 27 modes of vibrations account for the irreducible

Table 2. Energies of the anti-, transition state and syn-conformers and thermodynamic parameters of thiophene-2-carbonyl fluoride and selenophene-2-carbonyl fluoride

Y	Anti / Hartrees	Syn / Hartrees	Transition state/ Hartrees	B1/ kJ/mol	ΔE/ kJ/mol	ΔH/ kJ/mol	ΔG/ kJ/mol
DFT/6-311++G(d,p)							
S	-765.725585 (0.068659)	-765.725920 (0.068724)	-765.710483 (0.068117)	39.65	-0.88	-0.54	-0.40 (54.1)*
Se	-2769.053128 (0.067411)	-2769.053365 (0.067453)	-2769.038421 (0.066897)	38.61	-0.62	-0.54	-0.45 (54.6)
MP2/6-311++G(d,p)							
S	-763.010428 (0.068082)	-764.233464 (0.067901)	-762.995765 (0.068138)	38.50	-0.77	-3.30	2.04 (21.5)
Se	-2766.477456 (0.066993)	-2766.477869 (0.066985)	-2766.465136 (0.066863)	32.34	-1.08	-1.04	-1.29 (72.7)
CCSD(T) /6-311++G(d,p)// MP2/6-311++G(d,p)							
S	-763.010428	-763.011126	-762.995765	38.50	-1.83		
Se	-2765.263723	-2765.264482	-2765.249672	36.89	-1.99		

* Percentage of the syn conformer at 298.15 K

representations $\Gamma_v = 8A'' + 19A'$ of the C_s point group of the conformers. The infrared vibrational frequency and Raman activity are dominated by the high intensity of the carbonyl stretching frequency. The transition states have been confirmed by the one and only one negative frequency. The imaginary frequencies (cm^{-1}) for (**A**) and (**B**) computed at MP2 level are -88.83 and -81.02 and at B3LYP level are -90.49 and -84.58.

Table 3. Some infrared frequencies, their intensities, Raman activities and assignments for the syn and anti conformers of the title compounds computed at B3LYP/6-311++G(d,p) [*Values in bracket are those from MP2 computations]

Frequencies/ cm^{-1}	IR Intensity/ km mol^{-1}	Raman Activity/ $Å^4$ amu^{-1}	Symmetry	Assignment
Syn thiophene-2-carbonyl fluoride				
84.5 (80.6)*	0.1 (0.1)	0.3	A"	Twisting $O_5C_2F_{11}$
1860.1 (1883.2)	526.3 (293.2)	155.0	A'	Stretching C_2O_5
755.8 (784.0)	4.1 (1.5)	4.9	A'	Scissoring $C_1C_4C_6$
725.1 (742.2)	2.5 (6.4)	24.5	A'	Scissoring $F_{11}C_2O_5$
645.1 (653.4)	9.5 (10.4)	3.9	A'	Scissoring $C_1S_3C_8$
463.5 (468.6)	0.1 (0.3)	3.3	A'	Scissoring $C_2C_1C_4$
365.1 (374.2)	2.9 (2.6)	3.7	A'	Bending $O_5C_2F_{11}$
178.0 (181.0)	1.6 (1.4)	0.2	A'	Bending C_1C_2
Syn selenophene-2-carbonyl fluoride				
405.2 (371.9)	1.8 (2.6)	0.6	A"	Wagging $C_1Se_3C_8$
78.4 (59.0)	0.1 (0.1)	0.1	A"	Twisting $O_5C_2F_{11}$
1852.3 (1853.8)	508.5 (353.2)	152.7	A'	Stretching C_2O_5
942.5 (967.4)	210.5 (201.5)	4.0	A'	Stretching C_2F_{11}
699.9 (710.4)	8.3 (15.3)	20.5	A'	Scissoring $O_5C_2F_{11}$
565.2 (568.6)	3.7 (2.4)	12.2	A'	Bending $C_1C_2O_5$
389.7 (391.8)	0.2 (0.2)	3.4	A'	Bending $C_1Se_3C_8$
327.6 (334.0)	1.9 (1.8)	3.8	A'	Bending $C_2C_1C_4$
152.4 (156.1)	2.2 (1.7)	0.3	A'	Bending C_1C_2

Table 3. *(Continued)*

Trans thiophene-2-carbonyl fluoride				
454.5 (403.2)	0.8 (0.3)	0.5	A"	Wagging $C_1S_3C_8$
80.3 (58.2)	0.5 (0.2)	0.8	A"	Twisting $O_5C_2F_{11}$
1862.5 (1869.4)	533.0 (400.0)	165.7	A'	Stretching C_2O_5
991.4 (1014.0)	170.7 (165.3)	1.7	A'	Stretching C_2F_{11}
708.0 (724.0)	21.3 (20.8)	19.4	A'	Scissoring $O_5C_2F_{11}$
448.7 (458.1)	0.2 (0.1)	3.1	A'	Bending $C_2C_1S_3$
383.8 (385.9)	4.2 (3.7)	4.5	A'	Bending $O_5C_2F_{11}$
171.5 (175.8)	0.9 (0.8)	0.2	A'	Bending C_1C_2
Trans selenophene-2-carbonyl fluoride				
734.7 (713.4)	55.3 (92.6)	0.1	A"	Bending C_1C_2
407.4 (379.8)	2.1 (3.0)	1.0	A"	Wagging $C_1S_3C_8$
73.9 (60.2)	0.5 (0.3)	0.7	A"	Twisting $O_5C_2F_{11}$
1858.4 (1864.3)	538.1 (403.5)	175.9	A'	Stretching C_2O_5
971.4 (991.2)	161.5 (146.1)	2.6	A'	Stretching C_2F_{11}
691.2 (702.4)	30.8 (34.5)	15.5	A'	Scissoring $O_5C_2F_{11}$
380.9 (389.2)	1.0 (0.8)	4.7	A'	Bending $C_1C_2F_{11}$
352.8 (352.9)	3.1 (2.8)	4.0	A'	Bending C_2O_5
148.5 (151.4)	0.9 (0.7)	0.3	A'	Bending C_1C_2

Table 4. Solvent effect on rotational barriers, energy differences and thermodynamic parameters

Solvent	Dielectric constant	B1/ kJ/mol	ΔE/ kJ/mol	ΔH/ kJ/mol	ΔG/ kJ/mol	% of trans conformer at 298.15 K
Sulfur analogue						
Heptane	1.92	40.38	-0.68	-0.56	-0.83	58.31
Chloroform	4.90	41.00	-0.77	-0.54	-0.34	53.46
Tetrahydofuran	7.58	41.18	-0.80	-0.55	0.10	48.97
Dichloroethane	10.36	41.26	-0.82	-0.58	-0.31	53.17
Acetone	20.70	41.33	-0.87	-0.71	-0.49	54.88
Ethanol	24.55	41.30	-0.88	-0.63	-0.12	51.16
Methanol	32.63	41.21	-0.91	-0.66	0.52	44.78
Dimethylsulfoxide	46.70	41.42	-0.89	-0.64	-0.09	50.95
Water	78.39	41.24	-0.90	-0.80	0.29	47.10
Selenium analogue						
Heptane	1.92	38.91	-0.59	-0.51	-0.45	54.52
Chloroform	4.90	39.11	-0.59	-0.51	-2.19	70.77
Tetrahydofuran	7.58	39.17	-0.60	-0.43	-2.01	69.26
Dichloroethane	10.36	39.13	-0.60	-0.56	-2.64	74.33
Acetone	20.70	39.13	-0.62	-0.16	0.10	48.97
Ethanol	24.55	39.09	-0.62	-0.36	0.005	49.95
Methanol	32.63	39.13	-0.62	-0.42	-1.00	59.97
Dimethylsulfoxide	46.70	39.14	-0.63	-0.54	-1.99	69.06
Water	78.39	38.79	-0.62	-0.45	0.75	42.48

It is found that there are systematic changes in the structures of the conformers of the (**A**) and (**B**) when they are studied in different solvents but these are not significant. To be more precise, the largest change in bond length is 0.012 Å and the largest change in bond angle is 1.0°. Similarly there are only small changes in the infrared vibrational frequencies. The most apparent effect of the solvents is that due to solute-solvent interaction, stabilization of the conformers depends on the dipole moments. The net effect of the solvent is that energy difference between the conformers decreases but rotational barrier increases. However an increase in the polarity of the solvent leads to a leveling effect of the different parameters. The solvent effect on rotational barriers, energy differences, enthalpies and free energy changes are summarized in Table 4.

4 Conclusion

This work reports a systematic investigation of the syn-, transition state and anti-conformers of thiophene-2-carbonyl fluoride (**A**) and selenophene-2-carbonyl fluoride (**B**) in the gas and solution phases. Some of the results for (**A**) compare satisfactorily with literature and the results for (**B**) should be helpful for reference, as it has not been studied previously. An interesting outcome of this work is that fluorine is strongly electronegative and thus can affect conformational equilibrium [18]. However in this case, replacing hydrogen of the carbonyl group by fluorine does result in a change in conformational preference relative to the parent compounds and this can be explained on the basis of the opposite charges on sulfur or selenium and oxygen in (**A**) and (**B**). Hence the syn conformer to be more preferred for both (**A**) and (**B**).

Acknowledgments. The authors acknowledge facilities from the Universiti Sains Malaysia and the University of Mauritius. The authors are also grateful to anonymous referees for their useful comments in improving the manuscript.

References

1. Han, I.-S., Kim, C.K., Jung, H.J., Lee, I.: Ab Initio Studies on the Rotational Equilibria of 2-Substituted Furan and Thiophene Carbonyl Derivatives. Theor. Chim. Acta. 93, 199–210 (1996)
2. Banks, J.W., Batsanov, A.S., Howard, J.A.K., O'Hagan, D., Rzepa, H.S., Martin-Santamaria, S.: The Preferred Conformation of α-Fluoroamides. J. Chem. Soc. Perkin Trans. 2, 2409–2411 (1999)
3. Abraham, R.J., Bretschneider, E., Orville-Thomas, W.J.: Internal Rotation in Molecules, ch. 13. Wiley, London
4. Ashish, H., Ramasami, P.: Rotational Barrier and Thermodynamical Parameters of Furfural, Thiofurfural, and Selenofurfural in the Gas and Solution Phases: Theoretical Study Based on Density Functional Theory Method. Mol. Phys. 106, 175–185 (2008)
5. Braathen, G.O., Kveseth, K., Nielsen, C.J.: Molecular Structure and Conformational Equilibrium of Gaseous Thiophene-2-aldehyde as Studied by Electron Diffraction and Microwave, Infrared, Raman and Matrix Isolation Spectroscopy. J. Mol. Struc. 145, 45–68 (1986)

6. Concistrè, M., Luca, G.D., Longeri, M., Pileio, G., Emsley, J.W.: The Structure and Conformations of 2-Thiophenecarboaldehyde Obtained from Partially Average Dipolar Couplings. Chem. Phys. Chem. 6, 1483–1491 (2005)
7. Fleming, G.D., Koch, R., Vallette, M.M.C.: Theoretical Study of the Syn and Anti Thiophene-2-aldehyde Conformers using Density Functional Theory and Normal Coordinate Analysis. Spectrochim. Acta A 65, 935–945 (2006)
8. Gronowitz, S.: Selenophene, a Twin-Brother of Thiophene? Phosphorus Sulfur 136, 59–90 (1998)
9. Salatelli, E., Zanirato, P.: The Conversion of Furan-, Thiophene- and Selenophene-2-carbonylazides into Isocyanates: A DSC Analysis. Arkivoc xi, 6–16 (2002)
10. Kim, H., Yoon, Y.-J., Kim, H., Cha, E.-Y., Lee, H.S., Kim, J.-H., Yi, K.Y., Lee, S., Cheon, H.G., Yoo, S.-E., Lee, S.-S., Shin, J.-G., Li, K.-H.: Vitro Metabolism of a New Cardioprotective Agent. KR-33028 in the Human Liver Microsomes and Cryopreserved Human Hepatocytes 28, 1287–1292 (2005)
11. Frisch, M.J., et al.: Gaussian 03, Revision B04. Gaussian Inc., Wallingford (2004)
12. Dennington II, R., Keith, T., Millam, J., Eppinnett, K., Hovell, W.L., Gilliland, R.: GaussView, Version 3.09. Semichem, Inc., Shawnee Mission (2003)
13. Mennucci, B., Tomasi, J.: Continuum Solvation Models: A New Approach to the Problem of Solute's Charge Distribution and Cavity Boundaries. J. Chem. Phys. 106, 5151–5158 (1997)
14. Cancès, E., Mennucci, B., Tomasi, J.: A New Integral Equation Formalism for the Polarizable Continuum Model: Theoretical Background and Applications to Isotropic and Anisotropic Dielectrics. J. Chem. Phys. 107, 3032–3041 (1997)
15. Mennucci, B., Cancès, E., Tomasi, J.: Evaluation of Solvent Effects in Isotropic and Anisotropic Dielectrics and in Ionic Solutions with a Unified Integral Equation Method: Theoretical Bases, Computational Implementation, and Numerical Applications. J. Phys. Chem. B 101, 10506–10517 (1997)
16. Tomasi, J., Mennucci, B., Cancès, E.: The IEF Version of the PCM Solvation Method: An Overview of a New Method Addressed to Study Molecular Solutes at the QM Ab Initio Level. J. Mol. Struct. (Theochem.) 464, 211–226 (1999)
17. Ramasami, P.: Theoretical Study of 2-Selenophenecarboaldehyde in the Gas and Solution Phases: Rotational Barrier, Energy difference and Thermodynamics Parameters. Communicated
18. Abdallah, H.H., Ramasami, P.: Rotational Barrier, Energy difference and Thermodynamical Parameters of 2-Furoylfluoride, its Sulfur and Selenium Analogues: Theoretical Study in the Gas and Solution Phases. Communicated

Density Functional Calculation of the Structure and Electronic Properties of Cu_nO_n (n=1-4) Clusters

Gyun-Tack Bae and Randall W. Hall

Department of Chemistry, Louisiana State University, Baton Rouge, Louisiana 70808

Abstract. We have performed *ab initio* Monte Carlo simulated annealing simulations and density functional theory calculations to study the structures and stabilities of copper oxide clusters, Cu_nO_n (n=1-4). We determined the lowest energy structures of neutral, positive and negatively charged copper oxide clusters using the B3LYP/LANL2DZ model chemistry. The geometries are found to undergo a structural change from two- to three-dimensions when $n = 4$ in the neutral clusters. We have investigated the size dependence of selected electronic properties of the binding energies, second differences of the energy, ionization potentials, electron affinities, and HOMO-LUMO gaps. We also have investigated fragmentation channels and charge distributions.

Keywords: Densitiy functional theory, copper oxide clusters.

1 Introduction

Studies of clusters help understand the evolution of properties from isolated atoms to bulk matter as well as to probe the details of solvation. Recently, it has been found that metal oxide clusters contribute to health hazards [1,2,3,4]. The high speed collision of clusters with solid surfaces can give rise to short-lived species at extreme temperature and pressures. It has been shown that these impact-heated clusters provide an environment in which chemical reactions can be induced. Energetic cluster impact also has the potential for technological application in the formation of particularly dense and coherent metal and semiconductor thin films. Metal oxide clusters formed during combustion react with many organic compounds. [5, 6] The practical uses of metal clusters are well known and include catalysis, nanomaterials, and composite materials.

Experimental [7,8,9,10] and computational [11,12,13,14] studies exist for some small copper oxide clusters (1-2 copper atoms.) Three isomers have been suggested as possible structures for CuO_2 [15]: bent CuOO (bent, C_s), linear OCuO, and C_{2v} OCuO. Evidence has been found both the bent CuOO [16,17,18,19] and the linear OCuO [15, 20, 21]. Vibrational frequencies have been calculated [20] for CuO_3, $OCuO_2^-$, and $Cu(O_3)^-$. Recently, the structures of CuO_4, CuO_5 [22] and neutral and negatively charged CuO_6 [23] were determined using plane-wave density functional theory. In addition, Cu_2O_x (x=1-4) have been studied using

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 122–130, 2009.

anion photoelectron spectroscopy and density functional calculations [24, 25]. In this paper, we investigate the electronic and geometric structures of neutral and charged copper oxide clusters ($(CuO)_n$ n=1-4).

2 Method of Calculation

2.1 *Ab Initio* Monte Carlo Simulation

The study of atomic and molecular clusters is hindered by the existence of multiple isomers for a given size cluster. Without an *a priori* knowledge of the global energy minima, the use of computer simulation methods, such as simulated annealing, often allow the location of global minima. For this reason, we performed *ab initio* simulated annealing Monte Carlo (MC) simulations (using Gaussian 03 and homegrown scripts) [26] to locate stable geometric structures for these clusters. The simulations used multiple starting geometries for each cluster size. The temperature was decreased from 2000 K to 300 K over a period of up to 500 MC steps. We used the B3LYP (Becke's 3-parameter exchange functional with Lee-Yang-Parr correlation energy functional) [27, 28, 29] version of DFT to calculate the energy.

2.2 Basis Sets

We evaluated several basis sets as candidates for our studies. These were 6-31G** [30, 31, 32], 6-31++G** [33, 30, 34, 35, 36, 37], 6-311G** [36, 38, 39, 40], 6-311++G** [35, 41], LANL2DZ [42, 43, 44], and DGDZVP [45, 46]. MC simulations were followed by standard geometry optimization using [47]. Calculations found the lowest energy clusters for Cu_2O_n $(n = 1 - 4)$ shown in Fig. 1. For Cu_2O_3 and Cu_2O_4, different isomers were found depending on basis set (Cu_2O_3-a: 6-31G**; Cu_2O_3-b: 6-31++G**, 6-311G**, 6-311++G**, LANL2DZ and DGDZVP; Cu_2O_4-a: 6-31++G**, 6-311++G**, LANL2DZ and DGDZVP; Cu_2O_4-b: 6-31G** and 6-311G** .) Experimental data of electron affinities of Cu_2O to Cu_2O_4 clusters is available [48]. A comparison of calculated and measured electron affinities are shown in Table 1. The best agreement with experimental data is found with the LANL2DZ basis set, which is therefore used in the remainder of this work.

Table 1. Electron affinity comparing basis sets with experimental data

	Electron Affinities (eV)						
	6-31G**	6-31++G**	6-311G**	6-311++G**	LANL2DZ	DGDZVP	EXP [48]
Cu_2O	0.94	1.27	0.14	1.24	1.15	1.15	1.10
Cu_2O_2	1.41	2.33	0.89	1.76	2.41	2.24	2.46
Cu_2O_3	2.35	2.65	1.67	3.09	3.25	3.08	3.54
Cu_2O_4	3.26	3.34	2.75	3.35	3.54	3.31	3.50

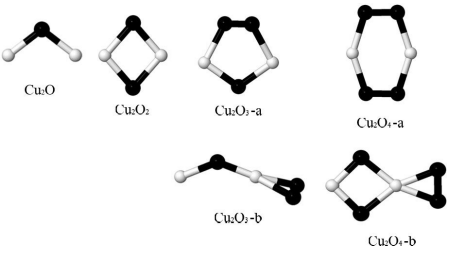


Fig. 1. Lowest energy clusters for Cu_2O_n, $n = 1 - 4$. Different basis sets give different lowest energy isomers for $n = 3$ and 4 (Cu_2O_3-a: 6-31G**; Cu_2O_3-b: 6-31++G**, 6-311G**, 6-311++G**, LANL2DZ and DGDZVP; Cu_2O_4-a: 6-31++G**, 6-311++G**, LANL2DZ and DGDZVP; Cu_2O_4-b: 6-31G** and 6-311G**)(see Table 1 for details.) White atoms are coppers and black atoms are oxygens.

3 Results and Discussion

3.1 Geometric Structure

The optimized structures of neutral, positive and negatively charged $(CuO)_n$ clusters with n=1-4 are shown in Fig. 2. The low-lying spin states (i.e., singlet, doublet, triplet, and quartet) of a given cluster were considered in the calculations. Every neutral copper oxide cluster, $(CuO)_n$, can be made from $Cu_{n-1}O_{n-1}$ cluster by attaching a Cu-O molecule to the side of $Cu_{n-1}O_{n-1}$ cluster.

In CuO, the Cu-O distances are 1.76Å(cation), 1.76Å(neutral) and 1.74Å(anion). Our calculated value of neutral Cu-O distance is in good agreement experimental value of 1.73 Å. [49] The spin states of optimized structures

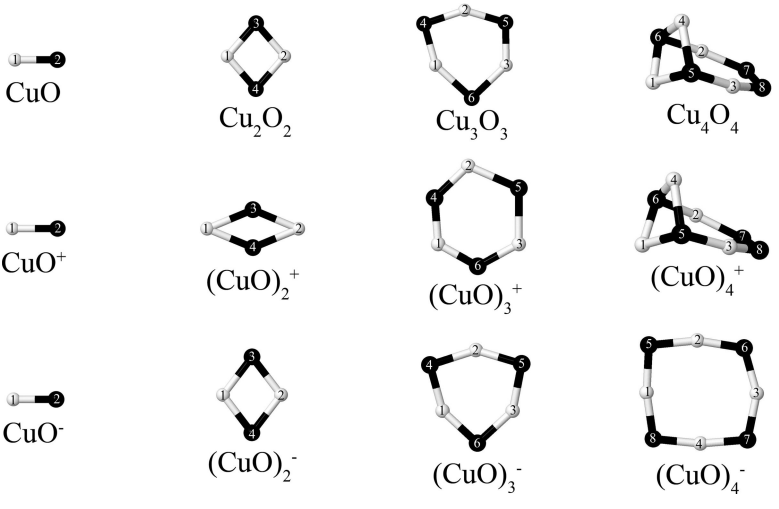


Fig. 2. Optimized structures of neutral, positively, and negatively charged $(CuO)_n$ clusters with n=1-4. White atoms are coppers and black atoms are oxygens.

are doublet, singlet and singlet for the neutral, cation and anion clusters, respectively. The structure of the lowest energy Cu_2O_2 cluster is rhombus in our simulations and calculations. The spin states of optimized structures are singlet, doublet, and doublet for the neutral, cation and anion clusters, respectively. Wang et al. [24] and Dai et al. [25] have suggested minimum energy structures for Cu_2O_2. Wang, et. al., suggest the structure is a rhombus while Dai, et. al, suggest the structure is linear or near linear. The Cu-O bond length of Cu_2O_2 structure found by Wang et al. is 1.78Å and the angle of Cu-O-Cu of Cu_2O_2 structure is 80°. Our Cu-O bond lengths are 2.01Å(cation), 1.86Å(neutral) and 1.92Å(anion) and average angle Cu-O-Cu of Cu_2O_2 structure is 81.2°. The Cu-O-Cu bond angles are 135°(cation), 82°(neutral), and 75°(anion). Cu_3O_3 clusters have nearly planar structures. The neutral cluster is a quartet while the charged clusters have triplet ground states. The average Cu-O-Cu bond angles are 121.8°(cation), 98.1°(neutral), and 94.2°(anion). The Cu-O-Cu bond angles are smaller than the O-Cu-O bond angles while the CuO are shorter than found in the Cu_2O_2 clusters. Our Cu-O bond lengths are 1.89Å(cation), 1.90Å(neutral) and 1.85Å(anion). The Cu_4O_4 cluster is the first nonplanar structure found for Cu_nO_n and consists of 2 copper atoms above and below the plane of a Cu_2O_4 unit. A similar structure is found for the cation cluster, while the anion cluster is planar. The spin states of optimized structures are triplet (neutral) and doublet (cation and anion.) Our Cu-O bond lengths are 1.92Å(cation), 1.94Å(neutral) and 1.81Å(anion).

3.2 Binding Energies, Ionization Potential, Electron Affinities

The binding energies per atom have been calculated from

$$E_b = [n\ E(Cu) + n\ E(O) - E(Cu_nO_n)]\,/2n \tag{1}$$

Table 2. Bond lengths (Å) of Cu-O in $(CuO)_n$ (n=1-4) clusters

	Clusters	d_{1-2}
CuO	CuO	d_{1-2}=1.81
	CuO^+	d_{1-2}=1.76
	CuO^-	d_{1-2}=1.74
Cu_2O_2	Cu_2O_2	d_{1-3}=1.86 d_{1-4}=1.86 d_{2-3}=1.86 d_{2-4}=1.86
	$Cu_2O_2^+$	d_{1-3}=2.01 d_{1-4}=2.01 d_{2-3}=2.01 d_{2-4}=2.01
	$Cu_2O_2^-$	d_{1-3}=1.92 d_{1-4}=1.92 d_{2-3}=1.92 d_{2-4}=1.92
Cu_3O_3	Cu_3O_3	d_{1-4}=1.83 d_{1-6}=2.06 d_{2-4}=1.81 d_{2-5}=1.83 d_{3-5}=1.83 d_{3-6}=2.03
	$Cu_3O_3^+$	d_{1-4}=1.75 d_{1-6}=1.77 d_{2-4}=1.78 d_{2-5}=2.13 d_{3-5}=2.12 d_{3-6}=1.79
	$Cu_3O_3^+$	d_{1-4}=1.84 d_{1-6}=1.85 d_{2-4}=1.85 d_{2-5}=1.85 d_{3-5}=1.85 d_{3-6}=1.85
Cu_4O_4	Cu_4O_4	d_{1-5}=1.96 d_{1-6}=1.97 d_{2-6}=1.88 d_{2-7}=1.93 d_{3-5}=1.88 d_{3-8}=1.93 d_{4-5}=1.97 d_{4-6}=1.96
	$Cu_4O_4^+$	d_{1-5}=1.94 d_{1-6}=1.93 d_{2-6}=1.88 d_{2-7}=1.94 d_{3-5}=1.87 d_{3-8}=1.93 d_{4-5}=1.92 d_{4-6}=1.95
	$Cu_4O_4^-$	d_{1-5}=1.81 d_{1-8}=1.81 d_{2-5}=1.80 d_{2-6}=1.79 d_{3-6}=1.81 d_{3-7}=1.83 d_{4-7}=1.80 d_{4-8}=1.80

Table 3. Spin states, ionization energies (IE), electron affinities (EA), and binding energies (E_b) for Cu_nO_n, $n = 1 - 4$. Energies are in electron volts and are calculated using the B3LYP/LANL2DZ model chemistry.

	Spin State	IE	EA	E_b
CuO	doublet			
CuO^+	singlet	12.25	1.35	1.22
CuO^-	singlet			
Cu_2O_2	singlet			
$Cu_2O_2^+$	doublet	8.24	2.35	1.85
$Cu_2O_2^-$	doublet			
Cu_3O_3	quartet			
$Cu_3O_3^+$	triplet	9.36	3.65	2.19
$Cu_3O_3^-$	triplet			
Cu_4O_4	triplet			
$Cu_4O_4^+$	doublet	8.37	3.40	2.35
$Cu_4O_4^-$	doublet			

Figure 3 shows the binding energy per atom, E_b, as a function of number of copper atoms in the cluster. There is a rapid increase from a binding energy of 1.24 eV (n = 1) to 2.22 eV (n = 3.) The n = 4 cluster has a similar binding energy to the n = 3 cluster, though not close to the bulk work function of >5 eV. Thus studies of larger clusters are necessary to more closely examine the size evolution of the properties of these clusters.

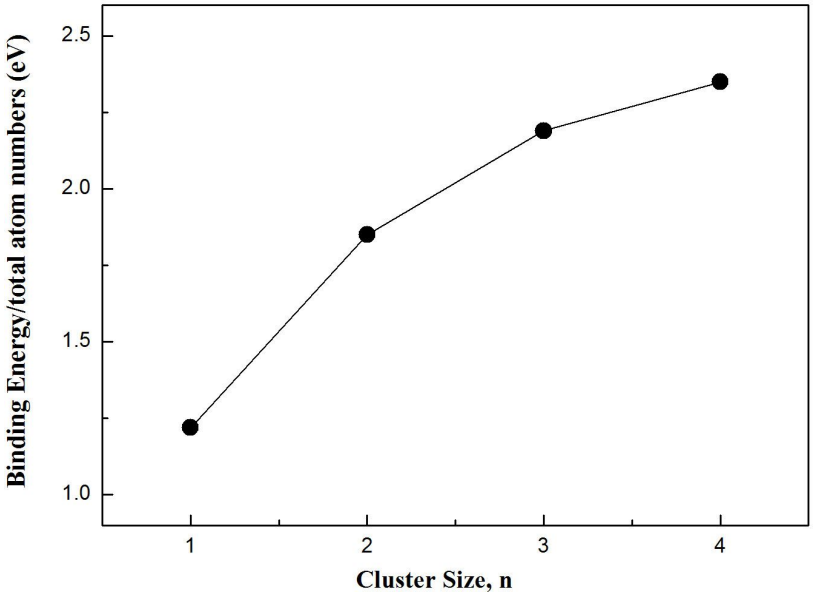


Fig. 3. Binding energies of neutral $(CuO)_n$ clusters with n=1-4

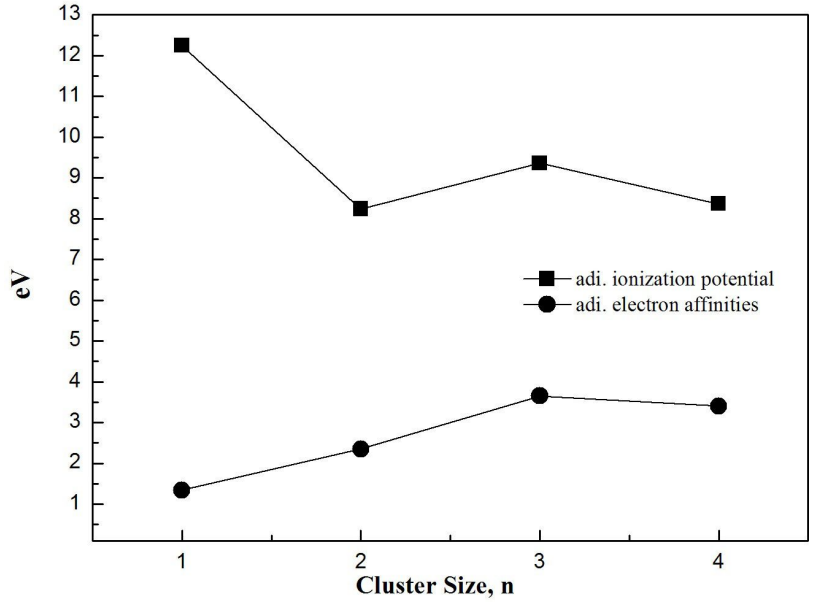


Fig. 4. Calculated adiabatic ionization potential and electron affinities of $(CuO)_n$ clusters with n=1-4

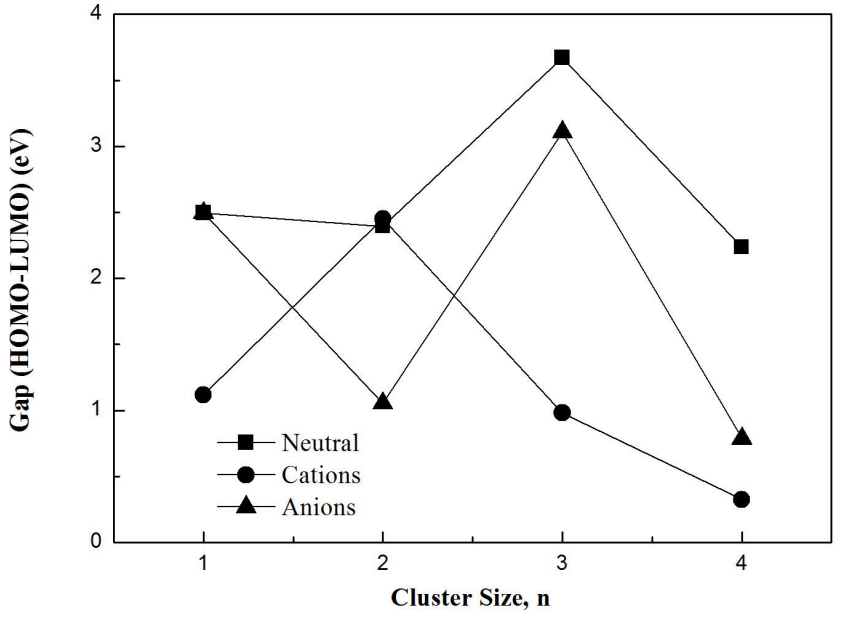


Fig. 5. Calculated HOMO-LUMO gap of $(CuO)_n$ clusters with n=1-4

Table 4. Fragmentation channels of $(CuO)_n$ clusters with n=1-4. Energies are in kcal/mol.

	ΔE
$(CuO)_2 \rightarrow CuO + CuO$	57.91
$(CuO)_3 \rightarrow Cu_2O_2 + CuO$	75.90
$(CuO)_4 \rightarrow Cu_3O_3 + CuO$	75.56
$\rightarrow Cu_2O_2 + Cu_2O_2$	93.55

Fig. 4 shows the adiabatic ionization potentials ($IP(X_n)=E(X_n^+)-E(X_n)$) and electron affinities ($EA(X_n)=E(X_n)-E(X_n^-)$). These properties display the even-odd oscillation often seen in clusters. The HOMO-LUMO gaps in these clusters are shown in Fig. 5.

3.3 Fragmentation Channels

We have also calculated the fragmentation energies of $(CuO)_n$ (n=1-4) clusters for various dissociation pathways. The fragmentation channels of $(CuO)_n$ clusters are shown in Table 4. The fragmentation energy of Cu_2O_2 cluster requires the least amount of energy to fragment and the lowest energy pathway for all clusters is to lose a single CuO group.

4 Conclusions

The electronic and structural properties of small copper oxide clusters have been studied using density functional theory and several basis sets. Comparison with existing experimental work demonstrated that the LANL2DZ basis set is in best agreement and therefore was used study study Cu_nO_n clusters. It was found that the clusters are planar for up to n = 3 and then become nonplanar. Ionization energies, electron affinities, and binding energies demonstrate some oscillations with cluster size, as is typical for clusters. Larger clusters must be studied in order to explore the approach to bulk properties.

Acknowledgements

This work was supported by NSF CBET-0625548 and CTS-0404314 grants and computational facilities at Louisiana State University (www.hpc.lsu.edu) and the Louisiana Optical Network Initiative (www.loni.org).

References

1. Pope III, C.A., Burnett, R.T., Thun, M.J., Calle, E.E., Krewski, D., Ito, K., Thurston, G.D.: JAMA 287, 1132 (2002)
2. Delfino, R.J., Gong, H., Linn, W.S., Pellizzari, E.D., Hu, Y.: Environ. Health Persp. 111, 647 (2003)

3. Donaldson, K., Li, X.Y., MacNee, W.: J. Aerosol Sci. 29, 553 (1998)
4. Air Quality Criteria for Particulate Matter 1-3, EPA/600/P-95/001 (1996)
5. Lighty, J.S., Veranth, J.M., Sarofim, A.F.: J. Air Waste Manage. Assoc. 50, 1565 (2000)
6. Linak, W.P., Wendt, J.O.L.: Fuel Process Technol. 39, 173 (1994)
7. De Heer, W.A.: Rev. Mod. Phys. 65, 611 (1993)
8. Morse, M.D.: Chem. Rev. 86, 1049 (1986)
9. Leopold, D.G., Ho, J., Lineberger, W.C.: J. Chem. Phys. 86, 1715 (1987)
10. Lee, T.H., Ervin, K.M.: J. Phys. Chem. 98, 10023 (1994)
11. Brack, M.: Rev. Mod. Phys. 65, 677 (1993)
12. Aakeby, H., Panas, I., Pettersson, L.G.M., Siegbahn, P., Wahlgren, U.: J. Phys. Chem. 94, 5471 (1990)
13. Calaminici, P., Kster, A.M., Russo, N., Salahub, D.R.: J. Chem. Phys. 107, 4066 (1996)
14. Cao, Z., Wang, Y., Zhu, J., Wu, W., Zhang, Q.: J. Phys. Chem. B 106, 9649 (2002)
15. Wu, H., Desai, S.R., Wang, L.S.: J. Chem. Phys. 103, 4363 (1995)
16. Kasai, P.H., Jones, P.M.: J. Phys. Chem. 90, 4239 (1986)
17. Mattar, S.M., Ozin, G.A.: J. Phys. Chem. 92, 3511 (1988)
18. Bauschlicher, C.W., Langhoff, S.R., Partridge, H., Sodupe, M.: J. Phys. Chem. 97, 856 (1993)
19. Hrusak, J., Koch, W., Schwarz, H.: J. Chem. Phys. 101, 3898 (1994)
20. Chertihin, G.V., Andrews, L., Bauschlicher, C.W.: J. Phys. Chem. A 101, 4026 (1997)
21. Deng, K., Yang, J., Yuan, L., Zhu, Q.: J. Chem. Phys. 111, 1477 (1999)
22. Massobrio, C., Pouillon, Y.: J. Chem. Phys. 119, 8305 (2003)
23. Pouillon, Y., Massobrio, C.: Chem. Phys. Lett. 356, 469 (2002)
24. Wang, L.S., Wu, H., Desai, S.R., Lou, L.: Phys. Rev. B 53, 8028 (1996)
25. Dai, B., Tian, L., Yang, J.: J. Chem. Phys. 120, 2746 (2004)
26. Frisch, M.J., Trucks, G.W., Schlegel, H.B., Scuseria, G.E., Robb, M.A., Cheeseman, J.R., Montgomery Jr, J.A., Vreven, T., Kudin, K.N., Burant, J.C., Millam, J.M., Iyengar, S.S., Tomasi, J., Barone, V., Mennucci, B., Cossi, M., Scalmani, G., Rega, N., Petersson, G.A., Nakatsuji, H., Hada, M., Ehara, M., Toyota, K., Fukuda, R., Hasegawa, J., Ishida, M., Nakajima, T., Honda, Y., Kitao, O., Nakai, H., Klene, M., Li, X., Knox, J.E., Hratchian, H.P., Cross, J.B., Bakken, V., Adamo, C., Jaramillo, J., Gomperts, R., Stratmann, R.E., Yazyev, O., Austin, A.J., Cammi, R., Pomelli, C., Ochterski, J.W., Ayala, P.Y., Morokuma, K., Voth, G.A., Salvador, P., Dannenberg, J.J., Zakrzewski, V.G., Dapprich, S., Daniels, A.D., Strain, M.C., Farkas, O., Malick, D.K., Rabuck, A.D., Raghavachari, K., Foresman, J.B., Ortiz, J.V., Cui, Q., Baboul, A.G., Clifford, S., Cioslowski, J., Stefanov, B.B., Liu, G., Liashenko, A., Piskorz, P., Komaromi, I., Martin, R.L., Fox, D.J., Keith, T., Al-Laham, M.A., Peng, C.Y., Nanayakkara, A., Challacombe, M., Gill, P.M.W., Johnson, B., Chen, W., Wong, M.W., Gonzalez, C., Pople, J.A.: Gaussian 03, Revision C.02. Gaussian, Inc., Wallingford CT (2004)
27. Lee, C., Yang, W., Parr, R.G.: Phys. Rev. B 37, 785 (1988)
28. Becke, A.D.: J. Chem. Phys. 98, 1372 (1993)
29. Stephens, P.J., Devlin, F.J., Chabalowski, C.F.: J. Phys. Chem. 98, 11623 (1994)
30. Francl, M.M., Petro, W.J., Hehre, W.J., Binkley, J.S., Gordon, M.S., DeFrees, D.J., Pople, J.A.: J. Chem. Phys. 77, 3654 (1982)
31. Rassolov, V.A., Pople, J.A., Ratner, M.A., Windus, T.L.: J. Chem. Phys. 109, 1223 (1998)

32. Hariharan, P.C., Pople, J.A.: Theo. Chim. Acta. 28, 213 (1973)
33. Hehre, W.J., Ditchfield, R., Pople, J.A.: J. Chem. Phys. 56, 2257 (1972)
34. Dill, J.D., Pople, J.A.: J. Chem. Phys. 62, 2921 (1975)
35. Clark, T., Chandrasekhar, J., Schleyer, P.v.R.: J. Comp. Chem. 4, 294 (1983)
36. Krishnam, R., Binkley, J.S., Seeger, R., Pople, J.A.: J. Chem. Phys. 72, 650 (1980)
37. Gill, P.M.W., Johnson, B.G., Pople, J.A., Frisch, M.: J. Chem. Phys. Lett. 197, 499 (1992)
38. Blaudeau, J.-P., McGrath, M.P., Curtiss, L.A., Radom, L.: J. Chem. Phys. 107, 5016 (1997)
39. Curtiss, L.A., McGrath, M.P., Blaudeau, J.-P., Davis, N.E., Binning Jr, R.C., Radom, L.: J. Chem. Phys. 103, 6104 (1995)
40. Glukhovtsev, M.N., Pross, A., McGrath, M.P., Radom, L.: J. Chem. Phys. 1995, 103 (1878)
41. Krishnan, R., Binkley, J.S., Seeger, R., Pople, J.A.: J. Chem. Phys. 72, 650 (1980)
42. Hay, P.J., Wadt, W.R.: J. Chem. Phys. 82, 270 (1985)
43. Wadt, W.R., Hay, P.J.: J. Chem. Phys. 82, 284 (1985)
44. Hay, P.J., Wadt, W.R.: J. Chem. Phys. 82, 299 (1985)
45. Godbout, N., Salahub, D.R., Andzelm, J., Wimmer, E.: Can. J. Chem. 70, 560 (1992)
46. Sosa, C., Andzelm, J., Elkin, B.C., Wimmer, E., Dobbs, K.D., Dixon, D.A.: J. Phys. Chem. 96, 6630 (1992)
47. Schmidt, M.W., Baldridge, K.K., Boatz, J.A., Elbert, S.T., Gordon, M.S., Jensen, J.H., Koseki, S., Matsunaga, N., Nguyen, K.A., Su, S.J., Windus, T.L., Dupuis, M., Montgomery, J.A.: GAMESS VERSION = 24 MAR 2007 (R3). J.Comput. Chem. 14, 1347 (1993)
48. Lou, L.: Phys. Rev. B 53, 8028 (1996)
49. Polak, M.L., Gilles, M.K., Ho, J., Lineberger, W.C.: J. Phys. Chem. 95, 3460 (1991)

Effects of Interface Interactions on Mechanical Properties in RDX-Based PBXs HTPB-DOA: Molecular Dynamics Simulations

Mounir Jaidann[1,2], Louis-Simon Lussier[1], Amal Bouamoul[1], Hakima Abou-Rachid[1], and Josée Brisson[2]

[1] Défense R & D Canada-Valcartier 2459 Boul. Pie-XI Nord, Québec, QC, Canada G3J 1X5
[2] CERMA and CQMF, Département de chimie, Faculté des sciences et de génie, Université Laval, Québec, Canada G1V 0A6
{Mounir.Jaidann,Louis-Simon.Lussier,Amal.Bouamoul, Hakima.Abou-Rachid}@drdc-rddc.gc.ca, Josee.Brisson@chm.ulaval.ca

Abstract. Atomistic molecular dynamics simulation was carried out to study interface interactions between a crystal structure and a plastic bonded explosive (PBX) system. In this work, the polymer is hydroxyl-terminated polybutadiene (HTPB), the plasticizer is dioctyl adipate (DOA) and the crystal phase is hexahydro-1,3,5-trinitro-1,3,5-triazine (RDX). Experimental RDX crystallo-graphic data show that (*020*), (*200*) and (*210*) crystal faces usually dominate, and these were therefore only these were studied. Interface models were built and interfacial bonding energies calculated to investigate HTPB/RDX adhesion properties in the (DOA+HTPB)/RDX system. Mechanical properties such as Poisson's ratio, Young, bulk and shear moduli were also predicted. The most favourable interactions occur between HTPB-DOA and the RDX (020) crystal face: obtaining crystals with prominent (020) faces may provide a more flexible mixture, with a lower Young's modulus and an increased ductility.

Keywords: Polymer-bonded explosives, molecular dynamics simulation, binding energy, mechanical properties.

1 Introduction

Plastic-bonded explosives (PBXs) are widely used in many defence and economic application because of their good safety, processing ease and high strength. The next generations of PBXs materials will possess improved insensitivity and energetic density while maintaining a good mechanical integrity. Atomistic molecular modeling may become a helpful tool in formulation conception, providing predictions on various properties of these systems. The use of modelling techniques can decrease hazards and accidents during formulation development, and contribute to minimizing the time-frame in which they can be screened and tested, eliminating poor formulations before even having to synthesize the compounds. Solid propellants and plastic bonded explosives incorporate various components, each having a specific function. First and

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 131–140, 2009.

foremost, a high density crystalline energetic material must be chosen, such as 1,3,5-triaza-1,3,5-trinitrocyclohexane (RDX), shown in Fig. 1, which has a high storage stability and is considered as the most powerful high explosive. The energetic material is incorporated in a polymeric binder, which provides acceptable mechanical properties and therefore greatly decreases the risk of accidental ignition during transportation, handling, and storage. The ability of a solid propellant to perform as an insensitive energetic material depends heavily on both the binder-plasticizer compatibility and explosives compatibility to binder and plasticizer. A binder such as hydroxyterminated polybutadiene (HTPB) requires a plasticizer with a similar chemical structure. A low weight aliphatic ester like dioctyladipate (DOA) is often used. With RDX as the main body (≈ 90% and above), PBXs contain a small amount of polymer/plasticizer (≈ 10%). Much attention has been paid to the explosives and polymers/plasticizer with respect to experimental measurement methods [1-6], as well as molecular dynamics (MD) simulation [7, 8]. However, few mechanical property simulations have been reported.

The HTPB-DOA system shown in Fig. 1 is a good model to test the use of modelling techniques for explosives, and has been the object of previous work in our group [9]. Crystalline surface morphology and crystal structure main planes existing at the crystal surface of RDX are known, and consist mainly in (*020*), (*200*) and (*210*) faces [10], depending on solvent. The present work will therefore focus on simulating the HTPB-DOA/RDX interface at these crystal faces. Simulations of crystal-amorphous phase have already been reported [7, 11, 12]. No investigation has however been performed in the presence of plasticizer, which is one of the key factors in PBXs stability and formulation.

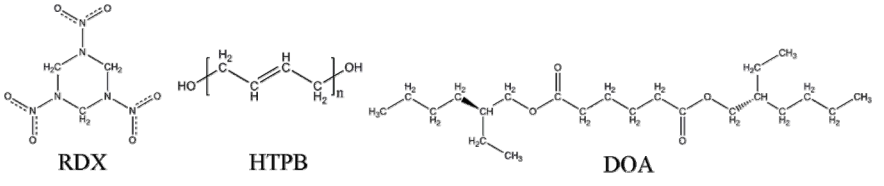


Fig. 1. Chemical structure of molecules in the studied system

Our main interest in the present work will therefore be to test the ability of atomistic methods to predict interactions between RDX and HTPB-DOA polymer-plasticizer system. Mechanical properties, interface between RDX crystal surface and the HTPB-DOA blend and interfacial bonding energy will also be investigated.

The final objective is to understand fundamental properties of RDX-based PBXs, which is of utmost importance in design and development of novel energetic materials. The present work will also allow investigation of the most interesting crystal morphology for RDX, interaction wise, and may allow for a better selection of experimental conditions (solvent, temperature, etc) in its processing prior to incorporation in PBXs. This work is part of a larger effort to evaluate composite systems prior to experimentally investigation, in search for new, more optimized explosives.

2 Computational Details

Molecular simulations were performed using the Materials Studio (MS) version 4.3 software commercialized by Accelrys. Molecular mechanics (MM) and molecular dynamics (MD) calculations were performed using the Discover module and the COMPASS (condensed-phase optimized molecular potentials for atomistic simulation studies) force field [13] under periodic boundary conditions. Coulombic interactions were described with the Ewald summation approach with an accuracy of 0.01 kcal/mol and an update width parameter of 1.0 Å. Crystal structure prediction was conducted with the MS Polymorph Predictor module. Crystal packings were generated randomly, energy minimized and subsequently ranked according to potential energy. For more details on Polymorph Predictor, the reader is referred to Gdanitz et al. [14] and Karfunkel et al. [15]. Amorphous polymer plasticizer and polymer-plasticizer phases consisting of hydroxyl-terminated polybutadiene and dioctyl adipate were created using a combination of the algorithm developed by Theodorou and Suter [16] and the scanning method of Meirovitch [17] implemented in the MS Amorphous Phase module. A single bond was added per step, under the single substate per state rule, while using a substate width of 20. A random number seed was used to insure a properly randomized distribution in the cell. Model systems thus built were submitted to a preliminary equilibration treatment consisting of 1000 steps using the canonical ensemble NVT (constant number of atoms, constant volume and constant temperature) dynamics simulation with a velocity rescaling method to maintain a constant temperature, followed by 100000 steps of minimization. Energy was minimized using three methods. The first used was steepest descent up to a maximum derivative of 100 kcal mol^{-1}, followed by the conjugate gradient method (using the Polak-Ribiere algorithm) down to maximum derivative of 10 kcal mol^{-1}. The Newton method using Broyden-Fletcher-Goldfarb-Shanno (bfgs) approach (the maximum derivative is 0.001 kcal mol^{-1}) was used last. Model building was followed by NVT dynamics simulation for 1.0 ns. Temperature in all simulations was equilibrated with the Andersen algorithm, using a collision rate of 1.0. The velocity Verlet algorithm was used to integrate equations of motion with a 1.0 fs time step. Only one blend composition was investigated, HTPB-DOA, containing one molecule of DOA and one 6-repeat unit HTPB chain, blend composition being of 49 w% HTPB and 51 w% DOA. Initial densities used for model building were experimental values 0.900 g cm^{-3} for HTPB and 0.927 g cm^{-3} for DOA. The density of amorphous blend system HTPB–DOA was calculated as the weight average of pure substance densities: initial HTPB–DOA density was estimated as 0.929 g cm^{-3}. The RDX-based PBX HTPB-DOA composition was based on 83 w% RDX crystal phase (16 RDX molecules) and 17 w% HTPB-DOA amorphous phase (6-repeat unit HTPB chains and one DOA molecule). Using the MS surface builder module, RDX surfaces were first prepared by cleaving the crystal phase at the desired surface plane (*hkl*), while insuring that the width and depth of the surface are larger than the non-bonded cut-off distance of 10.8Å. The cleaved surface was minimized as described above and was then placed in a supercell (2×2×1), over which the HTPB-DOA amorphous phase was inserted. The *c*-axis of the supercell was extended to 30Å, so that HTPB-DOA can 'see' only one side of the surface even under periodic boundary conditions. NVT molecular dynamics simulation was then performed for 200 ps with a 1.0 fs time step at 298 K,

followed by 50 ps production runs, during which data were collected for subsequent analysis.

3 Results and Discussion

In this work, the aim is to use modelling techniques for RDX-based PBXs HTPB-DOA and investigate the effect of interface interactions on mechanical properties. To evaluate the usefulness of the approach for real systems, for which crystal structure data are often unavailable, the RDX crystal structure was also predicted.

3.1 Crystal Structure Prediction

The optimized crystal parameters are reported in Table 1. A large number of structures are generated and ranked from the lowest total potential energy E_p to the highest. The lowest-energy corresponds to an orthorhombic Pbca space group, whereas the second lowest energy is monoclinic C2/c. In the seven lowest energy structures generated, density fluctuates from 1.75 to 1.99 g cm^{-3}. Large energy differences are observed, only the lowest being stable. Experimental data for the RDX crystal structure [18] is compared to the lowest-energy result in Table 2. RDX crystallizes in the orthorhombic space group, as found using Polymorph Predictor. Unit cell dimensions are in good agreement with experiment, *a*, *b* and *c* vector units showing discrepancies of 1.9%, -2.7% and -4.7%, respectively.

Table 1. Lowest-energy crystal structure models for RDX showing space group, density ρ (g/cm^3), potential energy E_p (kcal/mol), cell vectors a, b and c (Å) and angles α, β and γ (°)

Space group	ρ	E_p	a	b	c	α	β	γ
Pbca	1.91	-1611	13.43	11.26	10.21	90.00	90.00	90.00
C2/c	1.86	-1602	24.73	5.93	27.30	90.00	156.58	90.00
Pnma	1.75	-1592	10.66	26.05	6.07	90.00	90.00	90.00
$P2_12_12_1$	1.90	-806	5.87	15.38	8.59	90.00	90.00	90.00
$P2_1/c$	1.89	-806	6.62	11.20	10.63	90.00	83.33	90.00
$Pna2_1$	1.99	-798	7.21	10.85	9.46	90.00	90.00	90.00
$P2_1$	1.99	-399	6.51	9.46	6.51	90.00	67.24	90.00

The predicted density is 5.7% higher, which is reasonable considering the approximations made when using a force field. This excellent match clearly indicates that Polymorph Predictor, coupled to the COMPASS force field, is adequate to predict crystal structures of molecules representing similar chemical and conformational properties. The force field used should however always be evaluated for the chemical class under investigation. Crystal structures of molecules of higher flexibility will be increasingly difficult to predict, and therefore this approach can only be used, at the present time, for rigid molecules. For more flexible molecules, determining the experimental crystal structure is still essential.

Table 2. Properties of the RDX crystal: density (g/cm^3) and cell vectors (Å) ($\alpha = \beta = \gamma = 90°$)

RDX (Pbca)	ρ	a	b	c
Experimental [18]	1.82	13.18	11.57	10.71
Predicted	1.91	13.43	11.26	10.21

3.2 Simulation of the Crystalline RDX-Amorphous HTPB-DOA Supersystem

In the following simulation step, HTPB, DOA and HTPB-DOA amorphous phase models were built and optimized. HTPB-DOA models were selected using a criteria of low energy. This step was followed by construction of the crystalline RDX-amorphous HTPB-DOA supersystem, which is the most crucial step of this study, as it will allow interfacial interactions to be investigated. Further, mechanical properties will be estimated for an ideal 50:50 HTPB-DOA blend, for which no experimental mechanical properties have been reported in the literature. Similar modelling techniques have been used previously by various groups to simulate interactions in various PBXs systems [11]. It has been shown [8, 9] that DOA acts as a plasticizer in PBX formulations.

In the present work, an amorphous phase containing the polymer and the plasticiser was built using a concentration used in experimental formulations. It was however not possible to maintain a reasonable system size while strictly respecting the relative concentrations of amorphous and crystalline material. Instead, focus was put, when selecting the crystal-amorphous phase composition, on choosing a size which allowed reproduction of surface interactions with the plasticizer. Actual formulations are slightly richer in crystal phase than in plasticizer-polymer binding amorphous phase (90% crystal phase versus the modelled 83%). The simulated HTPB-DOA-RDX system is depicted in Fig. 2 for the three crystal surface planes considered. In the initial amorphous phase, HTPB and DOA were in a random coil-like conformation. This is clearly not the case after energy minimisation in presence of RDX: for all crystal cell orientations, HTPB and DOA molecules are now relatively extended, DOA is at the left in all three packing arrangements, and HTPB at the right side in the cell. Both are lying almost flat on top of the crystal structure, although some kinks are observable in both molecules above the (210) crystal cell. These chain arrangements are believed to allow a minimum in energy through intermolecular interactions. However, the occurrence of kinks in the (210) case qualitatively indicates that RDX/DOA and RDX/HTPB interactions are weaker in this case. Although DOA mainly acts as a HTPB plasticizer, due to the small concentration of the HTPB-DOA mixture both in the modelled systems (17 w%) and in typical experimental formulations (around 10%), DOA inevitably comes in close contact with the RDX crystalline phase and also forms strong van der Waals and electrostatic interactions.

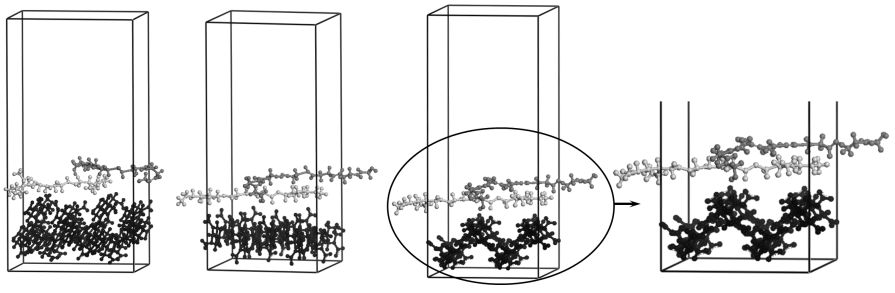


Fig. 2. Typical simulated HTPB-DOA/RDX supercells with (*210*) (*left*), (*200*) (*center*) and (*020*) (*right*) crystal surface, respectively, and zoom of the HTPB-DOA/RDX supercells with (*020*) crystal surface, (*HTPB in gray, DOA in light gray and RDX in black*)

Stability was checked by following potential and nonbond energy fluctuations. Systems were taken as stable and used for further calculations when these did not reach 10%. The interaction energy was calculated by using the following equation.

$$E_{Interaction} = E_{total} - \left(E_{surface} + E_{polymer}\right) \quad (1)$$

E_{total} is the energy of the surface and the polymer, $E_{surface}$ is the energy of the surface without the polymer and $E_{polymer}$ is the energy of the polymer without the surface. The binding energy E_{binding} reflects the intermolecular interactions between polymers and crystal, which is defined as the negative value of the interaction energy, which can be written as $E_{binding} = -E_{interaction}$. Results, reported in Table 3, vary with the crystallographic surface that is in contact with the polymer-plasticizer mixture. The (*020*) surface of RDX has the largest binding energies and therefore strongest ability to interact with the polymer, and the (*210*) surface has the smallest.

Table 3. Average binding energy (in kcal/mol) for different crystalline planes

(*hkl*) plane	E_{binding} (kcal/mol)
(*020*)	64
(*200*)	53
(*210*)	41

Pair correlation function g(r) was used to verify more objectively the degree of organization in these models [19]. Figure 3 shows g(r) values obtained after 250 ps NVT simulation of the three models. The three surfaces have comparable g(r) values at first glance, although maximum g(r) values are obtained for (*020*) and (*200*) surfaces, confirming the conclusion obtained from binding energies. In all cases, the largest peaks appear at r distances below 3.5 Å. For interatomic r distances higher than 3.5 Å, very few peaks are observed, and those which are have very small intensities, indicating that long-range order due to atom interactions is weak. These observations confirm that interactions at short distance play a role in system stability.

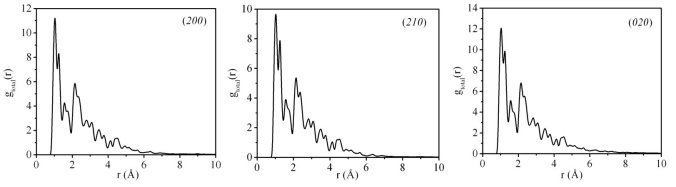


Fig. 3. Pair correlations functions total g(r) calculated for all atoms in HTPB-DOA/RDX models for different (*hkl*) planes, after 250 ps NVT dynamics simulation

Figure 4 shows the pair correlation function g(r) between H atoms of terminal HTPB hydroxyls and nitro oxygen atoms of RDX. The smallest distance d between H and O is observed for the HTPB-DOA/RDX (*020*) surface (d = 1.75 Å), which also exhibits the most intense g(r) peak, indicative of stronger and more numerous van der Waals and electrostatic interactions. This is in good correlation with the highest binding energy value observed for this model. In the case of the (*200*) and (*210*) crystallographic planes, peak distances vary between 2.05 Å and 2.15 Å respectively. Therefore, when HTPB comes in contact with a RDX crystal, the strongest interactions are observed with the (*020*) plane, and maximising the surface of this plane should improve adhesion between the crystalline phase and the polymer binder.

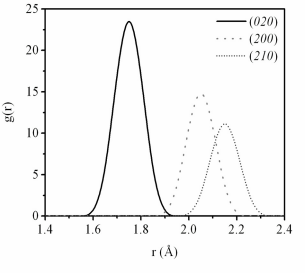


Fig. 4. Pair correlation function g(r) for H atom pairs of terminal HTPB hydroxyls and O atoms in RDX nitro groups for different (*hkl*) planes

3.3 Mechanical Properties

NPT (constant number of atoms, constant pressure and constant temperature) dynamics simulations were finally applied to these simulated systems to investigate mechanical properties of RDX-based PBXs. In this case the *c*-axis of the supercell is not extended. Calculations are based on a formalism due to Parrinello and Rahman [20] in which the elastic stiffness tensor C_{ij} is expressed in terms of fluctuations in the elastic strain tensor $\langle \varepsilon_i \varepsilon_j \rangle$ for the material. The NPT dynamics simulation was then performed for 200 ps with a time step of 1.0 fs at 298 K and a cell mass of 20.00 atomic mass units, followed by production runs of 100 ps, during which data were collected for subsequent analysis. In the theory of linear elasticity, the stress and strain tensors σ and ε are related to the elastic stiffness tensor C_{ij} by $\sigma_i = C_{ij} \times \varepsilon_j$ ($i, j = 1,2,\ldots,6$) [21].

For an isotropic material, the stiffness matrix may be described by specifying only Lamé coefficients λ and μ.Poisson's ratio ν and various moduli (Young's *E*, bulk *K*, shear *G*) can be calculated assuming the material to be isotropic. Elastic moduli may then be written in terms of the Lamé coefficients as follows [22]:

$$\nu = \frac{\lambda}{2(\lambda+\mu)}, \quad E = \mu\left(\frac{3\lambda+2\mu}{\lambda+\mu}\right), \quad K = \lambda + \frac{2}{3}\mu, \quad G = \mu \tag{2}$$

Poisson's ratio ν is the ratio of transverse contraction strain to longitudinal extension strain in the direction of the stretching force. Tensile modulus *E* is also known as the Young modulus, and is the tangent or secant modulus of elasticity of a material in tension. The bulk modulus (*K*) of a substance measures the resistance of the substance to uniform compression. It is defined as the pressure increase needed to cause a given relative decrease in volume. The shear modulus (*G*) is defined as the ratio of shear stress to the shear strain. The *K/G* indicates the extent of the plastic range, this ratio is associated with ductility in cases where *K/G* is high, and with brittleness when it is low. Mechanical properties thus estimated should be taken as approximations, due to the small system size and anisotropy as well as force field approximations, and should be used for comparison purposes only. Values are summarized in Table 4.

Table 4. Mechanical properties of RDX-Based PBXs (in GPa) for the various (*hkl*) surfaces

	(200)	(210)	(020)	RDX	Ref
Tensile modulus (*E*)	23	68	11	18	[23]
Poisson ratio (ν)	0.33	0.32	0.36	0.22	[23]
Bulk Modulus (*K*)	23	65	14	13	[24]
Shear modulus (*G*)	8.5	26	4	-	
K/G	2.7	2.6	3.5	-	

In Table 4, Poisson ratios are similar for all three models, but a 22% difference in modulus is noted compared to the experimental value reported for the RDX crystal, which confirms the semi-quantitative nature of properties estimated using this approach. All other calculated mechanical properties vary with (*hkl*) directions: tensile, bulk and shear moduli are lowest for (*020*) and highest for (*210*). Resistance to elastic deformation, expressed by the tensile modulus (*E*), has decreased for the (*020*) plane, indicating less rigidity in contrast to (*200*) and (*210*) planes. In other words, the HTPB-DOA/RDX (*020*) blend behaves more like a rubber and is more flexible. Finally, one of the most important parameters for explosive formulation is the *K/G* quotient, which is a measure of brittleness. It is highest for (*020*) and lower for (*210*) and (*200*), showing that ductility is better in the latter two cases. Interactions are less important, which explains the increased ductility.

4 Conclusions

The HTPB-DOA/RDX energetic blend system has been used to investigate the usefulness of atomistic modelling in the study and design of novel energetic materials. With the rigid RDX molecule, it was possible to predict, using the Polymorph Predictor module, the correct crystal space group, while unit cell dimensions and density showed a good match. This prediction approach may therefore be used for new energetic molecules for which no experimental data are available. Interactions between a crystalline, energetic molecule (RDX), a polymer binder (HTPB) and a plasticizer (DOA) were modelled under periodic boundary conditions following an approach previously proposed in the literature for crystalline-polymer interfaces [11]. The most favourable interactions occur between HTPB-DOA and the RDX (*020*) crystal face, and g(r) calculations showed that the main interaction at play was hydrogen bonding between the terminal hydroxyl group of HTPB and RDX oxygen atoms of nitro groups. It can be concluded that crystal morphology will have an effect on mechanical properties of the final blend systems, and that improvement in mechanical properties could be obtained by adjusting crystal growth conditions to favor large (*020*) crystal surfaces. Finally, Parrinello-Rahman calculations of isotropic moduli (Young's E, bulk K, shear G) for crystalline RDX-based PBXs show that the resistance to elastic deformation is decreased in the case of (*020*) as compared to (*210*) and (*200*). Hydrogen bonds and other interactions that occur most dominantly for the RDX (*020*) crystal face with HTPB-DOA therefore provide a more flexible mixture, lowering the modulus and increasing ductility. This result suggests that choosing appropriate conditions (solvent, temperature) to favor the growth of (*020*) crystal faces may improve mechanical properties of resulting PBX formulations. These properties will be explored in the future using experimental methods in our lab.

Acknowledgments. The authors gratefully thank scientists P. Lessard and P. Brousseau from DRDC Valcartier for the fruitful discussions on all experimental data of polymer and plasticizers used in this work. The authors also wish to acknowledge the support of the Centre de Bioinformatique et de Biologie Computationelle of Université Laval. Financial support for this work was provided by the NSERCC (National Science and Engineering Council of Canada).

References

1. Dagley, I.J., Parker, R.P., Jones, D.A., Montelli, L.: Simulation and Moderation of the Thermal Response of Confined Pressed Explosive Compositions. Combust. Flame. 106, 428–441 (1996)
2. Scholtes, J.H.G.: Onderzoek Naar de Variatie van Mechanische Eigenschappen van HTPB PBX's. TNO-PML 1997-A55 report (1997)
3. Yoo, C.-S., Cynn, H., Howard, W.M., Holmes, N.: Equations of State of Unreacted High Explosives at High Pressures. In: 11th International Detonation Symposium Snowmass Village, CO, USA (1998)
4. Provatas, A.: Formulation and Performance Studies of Polymer Bonded Explosives (PBX) Containing Energetic Binder Systems. Part 1, Technical Report DSTO-TR-1397 (2003)

5. Siviour, C.R., Gifford, M.J., Walley, S.M., Proud, W.G., Field, J.E.: Particle Size Effects on the Mechanical Properties of a Polymer Bonded Explosive. J. Mater. Sci. 39, 1255–1258 (2004)
6. Doherty, R.M., Watt, D.S.: Relationship between RDX Properties and Sensitivity. Propellants, Explos. Pyrotech. 33, 4–13 (2008)
7. Sewell, T.D., Menikoff, R., Bedrov, D., Smith, G.D.: A Molecular Dynamics Simulation Study of Elastic Properties of HMX. J. Chem. Phys. 119, 7417–7426 (2003)
8. Jaidann, M., Abou-Rachid, H., Lafleur-Lambert, X., Lussier, L.-S., Gagnon, N., Brisson, J.: Modeling and Measurement of Glass Transition Temperatures of Energetic and Inert Systems. Polym. Eng. Sci. 48, 1141–1150 (2008)
9. Abou-Rachid, H., Lussier, L.-S., Ringuette, S., Lafleur-Lambert, X., Jaidann, M., Brisson, J.: On the Correlation between Miscibility and Solubility Properties of Energetic Plasticizers/Polymer Blends: Modeling and Simulation Studies. Propellants, Explos. Pyrotech. 33, 301–310 (2008)
10. ter Horst, J.H., Geertman, R.M., van der Heijden, A.E., van Rosmalen, G.M.: The Influence of a Solvent on the Crystal Morphology of RDX. J. Cryst. Growth. 198, 773–779 (1999)
11. Xiao, J., Huang, H., Li, J., Zhang, H., Zhu, W., Xiao, H.: A Molecular Dynamics Study of Interface Interactions and Mechanical Properties of HMX-based PBXs with PEG and HTPB. THEOCHEM. 851, 242–248 (2008)
12. Zhu, W., Xiao, J., Zhu, W., Xiao, H.: Molecular Dynamics Simulations of RDX and RDX-Based Plastic-Bonded Explosives. J. Hazard. Mater (2008), doi:10.1016/j.jhazmat.2008.09.021
13. Sun, H.: COMPASS: An ab Initio Force-Field Optimized for Condensed-Phase Applications-Overview with details on Alkane and Benzene Compounds. J. Phys. Chem. B. 102, 7338–7364 (1998)
14. Gdanitz, R.J., Karfunkel, H.R., Leusen, F.J.J.: The Prediction of Yet-Unknown Molecular Crystal Structures by Solving the Packing Problem. J. Mol. Graph. 11, 275–276 (1993)
15. Karfunkel, H.R., Rohde, B., Leusen, F.J.J., Gdanitz, R.J., Rihs, G.: Continuous Similarity Measure between Nonoverlapping X-ray Powder Diagrams of Different Crystal Modifications. J. Comput. Chem. 14, 1125–1135 (1993)
16. Theodorou, D.N., Suter, U.W.: Atomistic Modeling of Mechanical Properties of Polymeric Glasses. Macromolecules 19, 139–154 (1986)
17. Meirovitch, H.: Computer Simulation of Self-Avoiding Walks: Testing the Scanning Method. J. Chem. Phys. 79, 502–508 (1983)
18. Choi, C.S., Prince, E.: The Crystal Structure of Cyclotrimethylenetrinitramine. Acta Crystallogr. Sect. B. 28, 2857–2862 (1972)
19. Hansen, J.-P., McDonald, I.R.: Theory of Simple Liquids, 2nd edn. Academic Press, London (1990)
20. Parrinello, M., Rahman, A.: Strain Fluctuations and Elastic Constants. J. Chem. Phys. 76, 2662–2666 (1982)
21. Weiner, J.H.: Statistical Mechanics of Elasticity. John Wiley, New York (1983)
22. Mavko, G., Mukerji, T., Dvorkin, J.: The Rock Physics Handbook. Cambridge University, Cambridge (2003)
23. Annapragada, S.R., Sun, D., Garimella, S.V.: Prediction of Effective Thermo-Mechanical Properties of Particulate Composites. Comp. Mat. Sci. 40, 255–256 (2007)
24. Sewell, T.D., Bennett, C.M.: Monte Carlo Calculations of the Elastic Moduli and Pressure-Volume-Temperature Equation of State for Hexahydro-1,3,5-trinitro-1,3,5-triazine. J. Appl. Phys. 88, 88–95 (2000)

Pairwise Spin-Contamination Correction Method and DFT Study of MnH and H_2 Dissociation Curves

Satyender Goel[1] and Artëm E. Masunov[1,2]

[1] Nanoscience Technology Center, Department of Chemistry
[2] Department of Physics, University of Central Florida, 12424 Research Parkway, Suite 400, Orlando, FL32826, USA
amasunov@mail.ucf.edu

Abstract. A clear advantage of broken symmetry (BS) unrestricted density functional theory DFT is qualitatively correct description of bond dissociation process, but its disadvantage is that spin-polarized Slater determinant is no longer a pure spin state (a.k.a. spin contamination). We propose a new approach to eliminate the spin-contamination, based on canonical Natural Orbitals (NO). We derive an expression to extract the energy of the pure singlet state given in terms of energy of BS DFT solution, the occupation number of the bonding NO, and the energy of the higher state built on these bonding and antibonding NOs (as opposed to self-consistent Kohn-Sham orbitals). Thus, unlike spin-contamination correction schemes by Noodleman and Yamaguchi, spin-correction is introduced for each correlated electron pair individually and thus expected to give more accurate results. We validate this approach on two examples, a simple diatomic H_2 and transition metal hydride MnH.

1 Introduction

Difficulties in DFT description of the bond dissociation are routed in the fact that DFT was derived based on assumption of non-degenerate system [1, 2]. A clear advantage of unrestricted (also known as spin-polarized or broken spin-symmetry) solution is qualitatively correct description of bond dissociation process [3, 4]. Since exact exchange-correlation functional is not known, unrestricted Kohn-Sham (UKS) treatment improves approximate functionals by taking part of the static electron correlation into account. The situation can be seen as localization of α and β electrons on the left and right atoms of the dissociating bonds, respectively (left-right electron correlation). Broken symmetry (BS) UKS thus describes the transition from closed shell system to biradical smoothly, which is not possible with restricted open shell KS (ROKS) approach.

A disadvantage of UKS approach is that spin-polarized Slater determinant is no longer an eigenfunction of the spin operator. Hence, the average value of $<\hat{S}^2>$ is not, generally equal to the correct value of $S_z(S_z+1)$ [5]. Here S_z is ½ of the difference in total numbers of α and β electrons. This situation is known as spin contamination and $<\hat{S}^2>$ is often used as its measure. The common rule [6] is to neglect spin contamination if $<\hat{S}^2>$ differs from $S_z(S_z+1)$ by less than 10%. As a result of spin contamination, molecular geometry may be distorted toward the high-spin state one, spin density

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 141–150, 2009.

often becomes incorrect, and electron energy differs from the pure spin state ones. While some researchers argue that this spin contamination in DFT should be ignored [3] others recognize it as a problem affecting the energy. Possible solutions to spin contamination problem include constrained DFT [7, 8] and spin contamination correction schemes [9, 10], discussed below.

Heisenberg exchange coupling parameter J is often used to describe the difference in energy between the low and the high spin state. Positive value of *J* corresponds to ferromagnetic, and negative value corresponds to anti-ferromagnetic coupling. Since BS-DFT does not produce the energies of the pure spin states, the expression for *J* must account for spin contamination. The following expressions had been suggested for this purpose [11-14]:

$$J_1 = \frac{\left({}^{DFT}E_{BS} - {}^{DFT}E_T\right)}{S^2_{\max}}, J_2 = \frac{\left({}^{DFT}E_{BS} - {}^{DFT}E_T\right)}{S_{\max}(S_{\max}+1)}, J_3 = \frac{\left({}^{DFT}E_{BS} - {}^{DFT}E_T\right)}{\left\langle S^2\right\rangle_T - \left\langle S^2\right\rangle_{BS}} \quad (1)$$

Of these three, J_3 suggested by Yamaguchi can be reduced to J_1 and J_2 in the weak and strong limits, respectively.

A more complicated expressions for variable spin-correction, including dependence of J on overlap between corresponding spin polarized orbitals *p* and *q* were also derived recently [15, 16]. This approach was shown to result in more accurate *J* values for Cu^{2+} binuclear complexes [16, 17]. However, this variable spin-correction approach had not been applied to systems with two or more correlated electron pairs. In this contribution we apply spin correction approach to study two diatomics, a simple dihydride H_2 and transition metal hydride MnH.

2 Theory

Here we propose an alternative approach to variable spin-correction, based on canonical Natural Orbitals (NO) [18]. First, let us consider a diatomic system AB with one correlated electron pair, such as stretched H_2 molecule. We assume that restricted Kohn-Sham formalism yields higher energy for this system than unrestricted one, as the case of H_2 molecule far from equilibrium. Unrestricted KS description produces the natural orbitals *a, b* as eigenvectors of the total density matrix with the orbital occupation numbers n_a, n_b as corresponding eigenvalues. We further assume that $n_a < n_b$ which means that orbital *a* is antibonding, and orbital *b* is bonding NO. They are symmetry-adapted (*a* is Σ_u and *b* is Σ_g in case of H_2 molecule). Corresponding spin-polarized broken symmetry orbitals *p, q* can be expressed [19] as a linear combination of *a* and *b* using polarization parameter λ:

$$p = \frac{1}{\sqrt{1+\lambda^2}}(b + \lambda a); q = \frac{1}{\sqrt{1+\lambda^2}}(b - \lambda a) \quad (2)$$

This parameter is determined by the occupation numbers n_a and n_b as shown below. If alpha and beta electrons are localized on different parts of the molecule and do not overlap, the polarization parameter become unity and we arrive to Noodleman's weak interaction limit. In the general case of many-electron system the orbitals of the alpha set, besides being orthogonal to each other, are also orthogonal to the orbitals of

the beta set for a single exception of the corresponding beta orbital. The spin polarized orbitals obtained with the most standard quantum chemistry codes do not possess this property, which is why one has to produce the corresponding spin-polarized orbitals from NOs. BS solution can still be written as the Slater determinant in the basis of these corresponding orbitals as:

$$BS = 1/\sqrt{2}\left\|p_\alpha q_\beta\right\| = \frac{1}{\sqrt{2}}\left\|\begin{matrix} p_1\alpha_1 p_2\alpha_2 \\ q_1\beta_1 q_2\beta_2 \end{matrix}\right\| \tag{3}$$

Substitution of the corresponding orbitals from (2) into (3) separates the pure singlet and triplet components:

$$BS = \frac{1}{\sqrt{2}}\left\|\begin{matrix} p_1\alpha_1 p_2\alpha_2 \\ q_1\beta_1 q_2\beta_2 \end{matrix}\right\| = \frac{1}{\left(1+\lambda^2\right)}S + \frac{\lambda}{\left(1+\lambda^2\right)}T \tag{4}$$

$$BS = \frac{1}{\left(1+\lambda^2\right)}\left(b_1b_2 - \lambda^2 a_1a_2\right)\frac{\alpha_1\beta_2 - \beta_1\alpha_2}{\sqrt{2}} + \frac{\lambda}{\left(1+\lambda^2\right)}\left(a_1b_2 - b_1a_2\right)\frac{\alpha_1\beta_2 + \beta_1\alpha_2}{\sqrt{2}} \tag{5}$$

where indexes *1* and *2* mark coordinates of the electrons. The first term in this expression contains the linear combination of the two closed-shell singlets, the lower closed shell singlet S_1:

$$S_1 = b_1b_2\frac{\alpha_1\beta_2 - \beta_1\alpha_2}{\sqrt{2}} \tag{6}$$

and the higher closed shell singlet S_2:

$$S_2 = a_1a_2\frac{\alpha_1\beta_2 - \beta_1\alpha_2}{\sqrt{2}} \tag{7}$$

while the second term is proportional to one of the possible triplet states: $T = T_0\sqrt{2}$,

$$T_0 = \frac{\left(a_1b_2 - b_1a_2\right)}{\sqrt{2}}\frac{\alpha_1\beta_2 + \beta_1\alpha_2}{\sqrt{2}} \tag{8}$$

This triplet contribution is the reason why UKS solution is spin contaminated. Therefore, we are looking to extract the energy of the singlet term from BS energy E_{BS} using the energy of the triplet. The expectation value of Kohn-Sham operator $\hat{H}$ then becomes,

$$E_{BS} = \left\langle BS\middle|\hat{H}\middle|BS\right\rangle = \frac{1}{\left(1+\lambda^2\right)^2}\left\langle S\middle|\hat{H}\middle|S\right\rangle + \frac{\lambda^2}{\left(1+\lambda^2\right)^2}\left\langle T\middle|\hat{H}\middle|T\right\rangle + \frac{\lambda}{\left(1+\lambda^2\right)^2}\left(\left\langle S\middle|\hat{H}\middle|T\right\rangle + \left\langle T\middle|\hat{H}\middle|S\right\rangle\right) \tag{9}$$

The last two terms in (9) vanish out due to orthogonality of S and T states, introduced in (4).

$$\left\langle S \mid S\right\rangle = \left\langle b_1b_2 - \lambda^2 a_1a_2 \mid b_1b_2 - \lambda^2 a_1a_2\right\rangle = 1 + \lambda^4 \tag{10}$$

Using normalization condition and Substituting (10) into (4) one can obtain:

$$BS = \frac{\sqrt{1+\lambda^4}}{1+\lambda^2} \cdot S_0 + \frac{\lambda\sqrt{2}}{1+\lambda^2} \cdot T_0 \tag{11}$$

Where,

$$S_0 = \frac{S}{\sqrt{1+\lambda^4}} = \frac{1}{\sqrt{1+\lambda^4}}\left(S_1 + \lambda^2 S_2\right) \tag{12}$$

Hence the BS UKS energy can be written in terms of renormalized singlet and triplet S_0,T_0 as:

$$E_{BS} = \frac{1+\lambda^4}{\left(1+\lambda^2\right)^2}\left\langle S_0\middle|\hat{H}\middle|S_0\right\rangle + \frac{2\lambda^2}{\left(1+\lambda^2\right)^2}\left\langle T_0\middle|\hat{H}\middle|T_0\right\rangle \tag{13}$$

In non-relativistic case, the energy of the triplet T_0 is the same as the energy E_T for the single determinant triplet T_1=$a_1\alpha_1 \backslash b_2\alpha_2$;

$$E_T = \left\langle T_1\middle|\hat{H}\middle|T_1\right\rangle = \left\langle T_0\middle|\hat{H}\middle|T_0\right\rangle \tag{14}$$

Then the energy $E_{S'}$ of the pure singlet S_0 can be found from (14) as

$$E_{S_0} = \frac{\left(1+\lambda^2\right)^2}{1+\lambda^4} E_{BS} - \frac{2\lambda^2}{1+\lambda^4} E_T \tag{15}$$

This energy includes the non-dynamic electron correlation effects arising from the mixing of S_1 and S_2 states. In order to relate the polarization parameter λ to the occupation numbers n_a, n_b, we can expand the electron density matrix in the basis of *a* and *b* orbitals.

$$\rho(BS) = \begin{bmatrix} n_a & 0 \\ 0 & n_b \end{bmatrix}, \rho(S_1) = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}, \rho(S_2) = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}, \rho(T_0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{16}$$

From (11-12)

$$\rho(BS) = \frac{1}{1+\lambda^2}\rho(S_1) + \frac{\lambda^4}{\left(1+\lambda^2\right)^2}\rho(S_2) + \frac{2\lambda^2}{\left(1+\lambda^2\right)^2}\rho(T_0) \tag{17}$$

then

$$n_a = \frac{2\lambda^4}{\left(1+\lambda^2\right)^2} + \frac{2\lambda^2}{\left(1+\lambda^2\right)^2} = \frac{2\lambda^2}{1+\lambda^2} \tag{18}$$

$$n_b = \frac{2}{\left(1+\lambda^2\right)^2} + \frac{2\lambda^2}{\left(1+\lambda^2\right)^2} = \frac{2}{1+\lambda^2} \tag{19}$$

And finally

$$\lambda = \sqrt{2/n_b - 1} \tag{20}$$

$$E_{S_0} = \frac{4}{2n_b^2 + 4n_b + 4} E_{BS} - \frac{4n_b - 2n_b^2}{2n_b^2 + 4n_b + 4} E_T \tag{21}$$

Thus, for a system with one correlated electron pair one can obtain the pure singlet energy expressed in terms of energy of BS UKS solution, the occupation number of the bonding NO, and the energy of the triplet built on these bonding and antibonding NOs (as opposed to self-consistent KS orbitals). This expression is applicable to two –electron systems as well as to the systems which have in addition the unpolarized electron core or ferromagnetically coupled unpaired electrons. Extension to this technique to the case of several correlated electron pairs will be presented elsewhere. All systems, considered in this study were found to have only pair of fractionally occupied NOs, in addition to singly occupied and unpolarized MOs.

Most importantly our approach does not use spin operator for the correction; it considers natural occupancies. At present our approach is good to study spin contaminated systems with one correlated pair.

Thus, for a system with one correlated electron pair one can obtain the pure singlet energy expressed in terms of energy of BS UKS solution, the occupation number of the bonding NO, and the energy of the triplet built on these bonding and antibonding NOs (as opposed to self-consistent KS orbitals). This expression is applicable to two-electron systems as well as to the systems which have in addition the unpolarized electron core or ferromagnetically coupled unpaired electrons.

We will turn next to the systems with two correlated electron pairs, In that case (9) can be written as:

$$E_{BS} = \left\langle BS_1 \cdot BS_2 \middle| \hat{H} \middle| BS_1 \cdot BS_2 \right\rangle \tag{22}$$

Using (4),

$$BS_1 \cdot BS_2 = \left(\frac{\sqrt{1+\lambda_1^4}}{\left(1+\lambda_1^2\right)} S_{01} + \frac{\sqrt{2}\lambda_1}{\left(1+\lambda_1^2\right)} T_{01} \right) \left(\frac{\sqrt{1+\lambda_2^4}}{\left(1+\lambda_2^2\right)} S_{02} + \frac{\sqrt{2}\lambda_2}{\left(1+\lambda_2^2\right)} T_{02} \right) \tag{23}$$

$$= \frac{1}{\left(1+\lambda_1^2\right)\left(1+\lambda_2^2\right)} \left(\begin{array}{l} \sqrt{1+\lambda_1^4}\sqrt{1+\lambda_2^4} S_{01} S_{02} + \sqrt{2}\lambda_2 \sqrt{1+\lambda_1^4} T_{02} S_{01} + \\ \sqrt{2}\lambda_1 \sqrt{1+\lambda_2^4} T_{01} S_{02} + 2\lambda_1 \lambda_2 T_{01} T_{02} \end{array} \right) \tag{24}$$

Simplifying above eq. by replacing S_{01} and S_{02}:

$$S_{01} = \left(BS_1 - \frac{\sqrt{2}\lambda_1}{\left(1+\lambda_1^2\right)} T_{01} \right) \frac{\left(1+\lambda_1^2\right)}{\sqrt{1+\lambda_1^4}} \tag{25}$$

$$S_{02} = \left(BS_2 - \frac{\sqrt{2}\lambda_2}{\left(1+\lambda_2^2\right)} T_{02} \right) \frac{\left(1+\lambda_2^2\right)}{\sqrt{1+\lambda_2^4}} \tag{26}$$

We have

$$BS_1 \cdot BS_2 = \frac{1}{\left(1+\lambda_1^2\right)\left(1+\lambda_2^2\right)}\left(\begin{array}{l}\sqrt{1+\lambda_1^4}\sqrt{1+\lambda_2^4}S_{01}S_{02} + \sqrt{2}\lambda_2\left(1+\lambda_1^2\right)T_{02}BS_1 + \\ \sqrt{2}\lambda_1\left(1+\lambda_2^2\right)T_{01}BS_2 + 2\lambda_1\lambda_2 T_{01}T_{02}\end{array}\right) \tag{27}$$

Hence the BS UKS energy can be written in terms of renormalized singlet, triplet and mixture of triplet and BS state, $S_{01}S_{02}$, $T_{01}T_{02}$, $T_{02}BS_1$, $T_{01}BS_2$ as:

$$\begin{array}{l}E_{BS} = \frac{\left(1+\lambda_1^4\right)\left(1+\lambda_2^4\right)}{\left(1+\lambda_1^2\right)^2\left(1+\lambda_2^2\right)}\left\langle S_{01}S_{02}\left|\hat{H}\right|S_{01}S_{02}\right\rangle + \frac{2\lambda_2^2\left(1+\lambda_1^2\right)^2}{\left(1+\lambda_1^2\right)^2\left(1+\lambda_2^2\right)}\left\langle T_{02}BS_1\left|\hat{H}\right|T_{02}BS_1\right\rangle + \\ \frac{2\lambda_1^2\left(1+\lambda_2^2\right)^2}{\left(1+\lambda_1^2\right)^2\left(1+\lambda_2^2\right)}\left\langle T_{01}BS_2\left|\hat{H}\right|T_{01}BS_2\right\rangle - \frac{4\lambda_1^2\lambda_2^2}{\left(1+\lambda_1^2\right)^2\left(1+\lambda_2^2\right)}\left\langle T_{01}T_{02}\left|\hat{H}\right|T_{01}T_{02}\right\rangle\end{array} \tag{28}$$

Then the energy E_{SO} of the pure singlet $S_{01}S_{02}$ can be found from (28) as

$$\begin{array}{c}E_{BS}\cdot\left(1+\lambda_1^2\right)^2\left(1+\lambda_2^2\right)^2 = \left(1+\lambda_1^4\right)\left(1+\lambda_2^4\right)E_{S_0} + 2\lambda_2^2\left(1+\lambda_1^2\right)^2 E_{T_{02}BS_1} \\ + 2\lambda_1^2\left(1+\lambda_2^2\right)^2 E_{T_{01}BS_2} - 4\lambda_1^2\lambda_2^2 E_{T_{01}T_{02}}\end{array} \tag{29}$$

$$\begin{array}{c}E_{S_0} = \frac{1}{\left(1+\lambda_1^4\right)\left(1+\lambda_2^4\right)}\left(E_{BS}\cdot\left(1+\lambda_1^2\right)^2\left(1+\lambda_2^2\right)^2 - 2\lambda_2^2\left(1+\lambda_1^2\right)^2 E_{T_{02}BS_1}\right. \\ \left. - 2\lambda_1^2\left(1+\lambda_2^2\right)^2 E_{T_{01}BS_2} + 4\lambda_1^2\lambda_2^2 E_{T_{01}T_{02}}\right)\end{array} \tag{30}$$

Here we derive an expression to extract the energy of the pure singlet state from the energy of the broken symmetry DFT description of the low-spin state and energies of the high-spin states: pentuplet and two spin-contaminated triplets. Thus, unlike spin-contamination correction schemes by Noodleman [20] and Yamaguchi [13], spin-correction is introduced for each correlated electron pair individually and there fore is expected to give more accurate results.

3 Computational Details

We studied Potential Energy Curves (PEC) for hydrogen dimer H_2 and transition metal hydride MnH to validate the spin-contamination correction approach described above in section 2. MnH calculations were done with Gaussian03 [21] program using all-electron Wachters+f [22, 23] basis set. For H_2 we have used aug-cc-pVQZ basis set with CCSD and spin-polarized (unrestricted) DFT calculations.

Spin-correction described above in theory section is implemented as a combination of unix shell script and FORTRAN code. It reads Natural Orbitals (NO) printout from Gaussian03 job (keyword used was Punch=NO) and converts them into spin-polarized molecular orbitals. Script uses a threshold parameter to identify the correlated pair. The spin polarization of the electron core was neglected by adjusting the

threshold value to consider natural occupations integer. The provision is made for the spin-up orbital *p* to be the one largely localized on metal atom and, so that spin-down orbital q is predominantly localized on H atom. The new alpha orbital set is made of doubly occupied NOs, orbital *p*, singly occupied NOs, and weekly occupied NOs. The new beta orbital set was identical, except that *p* was replaced with *q*. These orbitals were further used to evaluate the energy with single SCF step and verify that it is close to BS energy obtained at self-consistence. The energy of the triplet is calculated with another single SCF step using the original NOs only. It was used by the script to extract the energy of the pure singlet. The keywords used for single SCF step with the modified orbital set were SCF (MaxCycle=1) and Guess=Cards.

4 Results and Discussion

Fig. 1 illustrates potential energy curves for H_2 with CCSD, BMK (uncorrected and corrected) and conventional Yamaguchi spin contamination correction based on S^2 value. One can see from the figure the difference appear at the shoulder of the potential energy surface, where uncorrected BMK curve significantly overestimate the energies. The corrected curve with our new spin-contamination correction code efficiently finds the point of difference and corrects the energy to give potential energy curve similar to that of wavefunction method CCSD. We have also plotted the conventional correction based spin operator by Yamaguchi et. al. for comparison purpose. Though both the corrections are equally good in predicting energies at the accuracy of wavefunction theory level CCSD, our approach is based on actual occupation nos., which would perform better when number of electron correlated pair will increase in the study. The further validation of the system with more than two electron correlation pairs will be discussed in future.

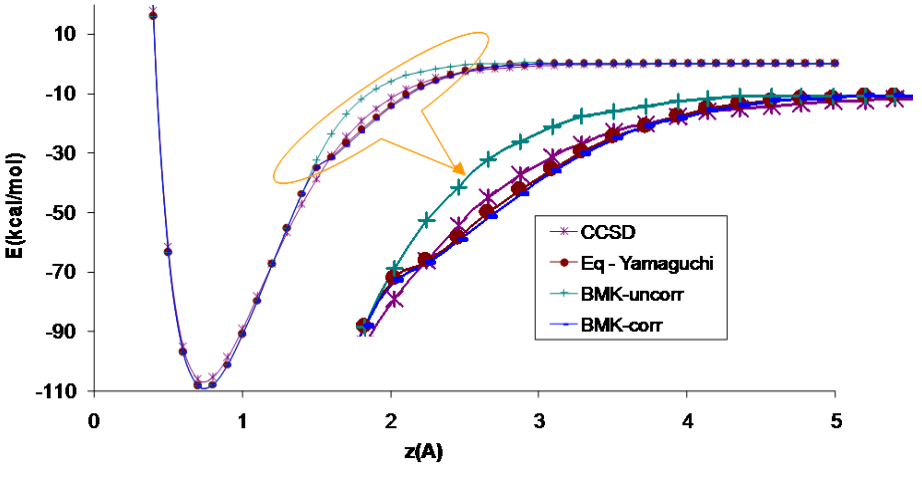


Fig. 1. Potential Energy Curves for hydrogen dimer with and without spin-contamination correction from our new approach, along with CCSD and Yamaguchi correction

In another attempt to check our new approach we considered more complex system MnH. Fig. 2 illustrates potential energy curve of two spin states of transition metal hydride, MnH with pure and hybrid DFT functionals TPSS and BMK. Our results are compared with PEC of only available WFT method MCSCF+SOCI in Fig 2 to equilibrium bond length for M=5. Table 1 shows the correction, introduced in Section 2, stabilizes this spin state by 3.1 kcal/mol below M=7, in agreement with experimental value reported in Borane et. al. [24] Thus, spin-corrected BMK predicts the ground state for MnH to have the multiplicity of 5 and accurately reproduces experimental D_e.

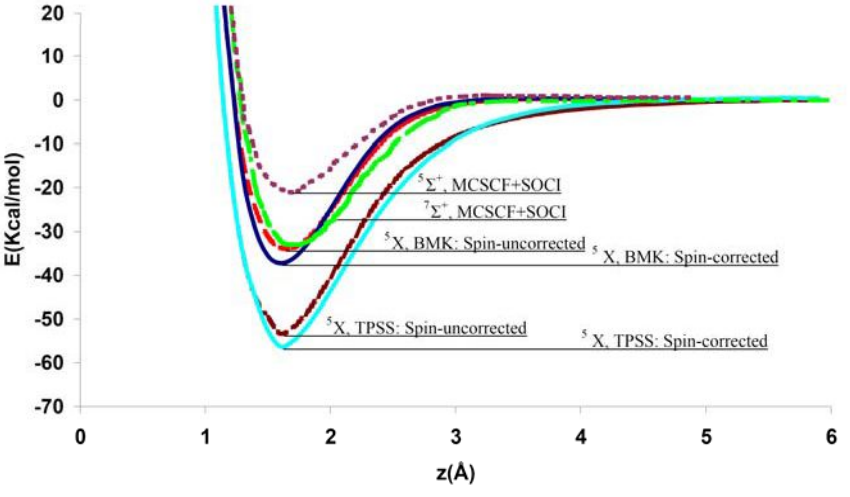


Fig. 2. Spin-corrected Potential Energy Curves of MnH with multiplicity 5 and 7, calculated by TPSS, BMK, and WFT (19) methods

Table 1. Spin corrected and uncorrected dissociation energies of MnH in Kcal/mol calculated with BMK and compared with *ab-initio* and experiment

	MnH
Multiplicity	**5**
BMK - Spin Uncorrected	**34.1**
BMK - Spin Corrected	**37.2**
MCSCF+SOCI[a]	**21.8**
Experiment[b]	**39.0**

[a][19], [b][24]

5 Conclusion

Here we derive an expression to extract the energy of the pure singlet state expressed in terms of energy of BS UKS solution, the occupation number of the bonding NO, and the energy of the triplet built on these bonding and antibonding NOs (as opposed to self-consistent KS orbitals). Thus, unlike spin-contamination correction schemes by Noodleman and Yamaguchi, spin-correction is introduced for each correlated electron pair individually and thus expected to give more accurate results. Diatomics considered for this study were found to have only pair of fractionally occupied NOs, in addition to singly occupied and unpolarized MOs. Our approach successfully predicts the correct spin state as validated by dihydrogen and manganese hydride in this study. This opens the venue to study more complicated enzymatic systems involving transition metals, more accurately with the help of DFT.

References

1. Hohenberg, P., Kohn, W.: Inhomogeneous Electron Gas. Phys. Rev. 136, B864–B871 (1964)
2. Kohn, W., Sham, L.J.: Self-Consistent Equations Including Exchange and Correlation Effects. Phys. Rev. 140, 1133 (1965)
3. Perdew, J.P., Savin, A., Burke, K.: Escaping the Symmetry Dilemma through a Pair-Density Interpretation of Spin-Density Functional Theory. Phys. Rev. A 51, 4531–4541 (1995)
4. Sherrill, C.D., Lee, M.S., Head-Gordon, M.: On the performance of density functional theory for symmetry-breaking problems. Chem. Phys. Lett. 302, 425–430 (1999)
5. Davidson, E.R., Clark, A.E.: Analysis of wave functions for open-shell molecules. Phys. Chem. Chem. Phys. 9, 1881–1894 (2007)
6. Young, D.: Computational Chemistry: A Practical Guide for Applying Techniques to Real World Problems, p. 408. Wiley-Interscience, Hoboken (2001)
7. Diaconu, C.V., et al.: Broken-symmetry unrestricted hybrid density functional calculations on nickel dimer and nickel hydride. J. Chem. Phys. 121, 10026–10040 (2004)
8. Wu, Q., Van Voorhis, T.: Direct optimization method to study constrained systems within density-functional theory. Phys. Rev. A. 72 (2005)
9. Lovell, T., et al.: A structural model for the high-valent intermediate Q of methane monooxygenase from broken-symmetry density functional and electrostatics calculations. J. Am. Chem. Soc. 124, 5890–5894 (2002)
10. Takeda, R., Yamanaka, S., Yamaguchi, K.: Resonating broken-symmetry approach to biradicals and polyradicals. Int. J. Quant. Chem. 106, 3303–3311 (2006)
11. Noodleman, L., Davidson, E.R.: Ligand Spin Polarization and Antiferromagnetic Coupling in Transition-Metal Dimers. Chem. Phys. 109, 131–143 (1986)
12. Bencini, A., et al.: Density functional modeling of long range magnetic interactions in binuclear oxomolybdenum(V) complexes. J. Phys. Chem. A 102, 10545–10551 (1998)
13. Yamaguchi, K., et al.: Extended Hartree-Fock (EHF) Theory of Chemical-Reactions.3. Projected Moller-Plesset (PMP) Perturbation Wavefunctions for Transition Structures of Organic-Reactions. Theo. Chim. Acta. 73, 337–364 (1988)
14. Yamanaka, S., et al.: Generalized spin density functional theory for noncollinear molecular magnetism. Int. J. Quan. Chem. 80, 664–671 (2000)

15. Neese, F.: Definition of corresponding orbitals and the diradical character in broken symmetry DFT calculations on spin coupled systems. J. Phys. Chem. Sol. 65, 781–785 (2004)
16. Ali, M.E., Datta, S.N.: Broken-symmetry density functional theory investigation on bis-nitronyl nitroxide diradicals: Influence of length and aromaticity of couplers. J. Phys. Chem. A. 110, 2776–2784 (2006)
17. Ali, M.E., Datta, S.N.: Theoretical investigation of magnetic properties of a dinuclear copper complex [Cu-2(mu-OAc)(4)(MeNHph)(2)]. J. Mol. Struc-Theochem. 775, 19–27 (2006)
18. Kozlowski, P.M., Pulay, P.: The unrestricted natural orbital-restricted active space method: methodology and implementation. Theo. Chem. Acc. 100, 12–20 (1998)
19. Chipman, D.M.: The Spin Polarization Model for Hyperfine Coupling-Constants. Theo. Chim. Acta. 82, 93–115 (1992)
20. Noodleman, L.: Valence Bond Description of Anti-Ferromagnetic Coupling in Transition-Metal Dimers. J. Chem. Phys. 74, 5737–5743 (1981)
21. Frisch, M.J.: GAUSSIAN 2003. 1994-2003. Gaussian Inc., Wallingford (2004)
22. Wachters, A.J.: Gaussian Basis Set for Molecular Wavefunctions Containing Third-Row Atoms. J. Chem. Phys. 52, 1033 (1970)
23. Hay, P.J.: Gaussian Basis Sets for Molecular Calculations - Representation of 3d Orbitals in Transition-Metal Atoms. J. Chem. Phys. 66, 4377–4384 (1977)
24. Barone, V., Adamo, C.: First-row transition-metal hydrides: A challenging playground for new theoretical approaches. Int. J. Quant. Chem. 61, 443–451 (1997)

Prediction of Exchange Coupling Constant for Mn_{12} Molecular Magnet Using Dft+U

Shruba Gangopadhyay[1,2], Artëm E. Masunov[1,2,3], Eliza Poalelungi[1,4], and Michael N. Leuenberger[1,3]

[1] Nanoscience Technology Center
[2] Department of Chemistry
[3] Department of Physics, University of Central Florida,
12424 Research Parkway, Suite 400, Orlando, FL 32826 USA
[4] Department of Physics, Alexandru Ioan Cuza University, Bulevardul Carol I, Nr.11, 700506, Iasi, Romania
amasunov@mail.ucf.edu, mleuenbe@mail.ucf.edu

Abstract. Single-molecule magnets are perspective materials for molecular spintronic applications. Predictions of magnetic coupling in these systems have posed a long standing problem, as calculations of this kind require a balanced description of static and dynamic electron correlation. The large size of these systems limits the choice of theoretical methods used. Two methods feasible to predict the exchange coupling parameters are broken symmetry Density Functional Theory (BSDFT) and DFT with empirical Hubbard U parameter (DFT+U). In this contribution we apply DFT+U to study Mn-based molecular magnets using Vanderbilt Ultrasoft Pseudopotential plane wave DFT method, implemented in Quantum ESPRESSSO code. Unlike most previous studies, we adjust U parameters for both metal and ligand atoms using two dineuclear molecular magnets $[Mn_2O_2(phen)_4]^{2+}$and $[Mn_2O_2(OAc)(Me_4dtne)]^{3+}$ as the benchmarks. Next, we apply this methodology to Mn_{12} molecular wheel. Our study finds antiparallel spin alignment in weakly interacting fragments of Mn_{12}, in agreement with experimental observations.

Keywords: DFT+U, Heisenberg exchange constant, Molecular magnet, Magnetic Wheel, molecular spintronics, quantum computing.

1 Introduction

Single molecule magnets (SMMs) have been of considerable interest to scientists ever since their initial discovery in 1993.[1, 2] SMMs are transition metal complexes that have a large spin ground state and considerable negative anisotropy leading to a barrier for the reversal of magnetization. These molecules show slow magnetization relaxation and can be magnetized below their blocking temperature.[3] The first SMM to be discovered was $[Mn_{12}O_{12}(CH_3COO)_{16}(H_2O)_4]_2CH_3COOH$, $4H_2O$, a dodecanuclear manganese cluster with a S=10 ground state, that is commonly known as Mn_{12}-acetate.[1, 2] This complex shows magnetization hysteresis and also shows quantum tunneling of the magnetization as evidenced by steps at regular intervals in the hysteresis loop.[3]

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 151–159, 2009.

Since the discovery of Mn_{12}-acetate, a large number of new SMMs have been reported with a wide variety of topologies and nuclearities, incorporating a variety of different metal atoms. A majority of molecules reported to show SMM behavior have been synthesized using manganese, iron or nickel. Manganese clusters that show SMM behavior are the most abundant. Many derivatives of Mn_{12} acetate have been reported to show SMM behavior. Examples of these include $[Mn_{12}O_{12}(O_2CCH_2Bu^t)_{16}(H_2O)_4]$ [4] and the mixed-carboxylate complex $[Mn_{12}O_{12}(O_2CCHCl_2)_8(O_2CCH_2Bu^t)_8(H_2O)_3]$, which were reported [5] to have an S=10 ground state. The complex $[Mn_{12}O_{12}(O_2CC_6H_4\text{-}2\text{-}CH_3)_{16}(H_2O)_4]$ $CH_2Cl_2{\cdot}2H_2O$ reported by Rumberger et al.,[6] is another SMM with Jahn-Teller isomerism. The complex $[Fe_8O_2(OH)_{12}(tacn)_6]Br_8$ is one of the most extensively studied iron SMMs; it has a S=10 ground state and incorporates the ligand triazocylcononane (tacn).

SMMs containing other transition metals such as cobalt or vanadium are relatively rare. An example of a cobalt SMM is the $[Co_4(hmp)_4(MeOH)_4Cl_4]$ complex, which has four Co (II) ions, was reported by Yang et al. and shows magnetization hysteresis at low temperatures.[7] The 2-hydroxymethylpyridine (hmp) ligand is a chelating ligand in this complex. Some of the different topologies seen in SMMs include Mn_4 dicubane [8] complexes and the S=9/2 Mn_4 cubane [9] complexes. Other interesting topologies include molecular wheels and rod-shaped SMMs such as the Mn_6 clusters. One-dimensional chains of weakly interacting SMMs are also known, such as the complex $[Mn_4(hmp)_6Cl_2]_n(ClO_4)_{2n}$, reported by Yoo et al.[10] Perhaps the most interesting of these topologies is that of the wheel-shaped SMMs. Scientists have been fascinated with molecular wheels for a number of reasons. Odd-numbered molecular wheels, such as $[(C_6H_{11})_2NH_2]{\cdot}[Cr_8NiF_9(O_2CC(CH_3)_3)_{18}]$ are of interest to scientists studying spin frustration.[14] One of the smallest of these molecular wheels is the recently reported tetranuclear manganese complex, with the formula $[Mn_4(anca)_4(Htea)_2(dbm)_2]2.5$ Et_2O.[11] Larger wheels include the Mn_{24} wheel, reported in 2006, which consists of eighteen Mn(III) ions and six Mn(IV) ions linked together to form a wheel-shaped topology.[12] It is believed that molecular wheels could be used in design of quantum computer.[13]

The family of wheel-shaped complexes that shows SMM behavior is steadily growing. Among these complexes are the Mn_{22} wheel [14] and the Mn_{84} wheel, which is the largest wheel-shaped SMM known to date. The largest reported spin ground state for a wheel-shaped SMM is the S=14 ground state reported for the complex $[Mn_{16}O_2(OCH_3)_{12}(tmp)_8(CH_3COO)_{10}]{\cdot}3Et_2O$.[15] Among the family of wheel-shaped SMMs is a smaller family of single-stranded wheels including the Mn_{16} wheel [23], which has the largest single-stranded loop known to date and was reported in 2005. A series of $[Mn_{12}]$ wheels reported by Rumberger et al. [6] in 2005 are also examples of single-stranded wheels.

In this contribution we predict Heisenberg exchange constant for Mn-based magnetic wheel using DFT+*U* method. We successfully validated the method for two Mn(IV) bimetallic system and then applied the same protocol to predict the value of the Heisenberg constant, which could not be predicted in the previous study using Hybrid Density functional theory.[16]

2 Computational Details

All the reported calculations were done using the PWSCF package,[17] which utilizes PBE exchange-correlation functional, Vanderbilt ultrasoft pseudopotentials [18] and a plane-wave basis set. The energy cutoffs for the wave functions and charge densities were set at 35 and 360 Ry to ensure total energy convergence. The Marzari-Vanderbilt [19] cold smearing with smearing factor 0.0008 was used for spin polarized calculation. All molecular structures were optimized in ferromagnetic state starting from atomic coordinates, obtained with X-Ray diffraction experiments. First we validated our method for two Mn(IV) complexes (I and II) and then applied to the Mn_{12} complex, referred as complex III. The DFT+*U* method described by Cao et. al.[20] was used for the calculations. We applied Hubbard *U* parameter on Mn atom, as well on the ligand oxygen and nitrogen atoms, coordinating the Mn atom. Since the Quantum-ESPRESSO code doesn't allow using *U* parameter on nitrogen, we modified the source code accordingly. Self-consistent Hubbard-*U* method has been incorporated to determine the *U* value for Mn which turns out to be 2.6 eV for this system. For oxygen and nitrogen we used the *U* values of 1.50 eV. Local Thomas-Fermi mixing mode was used to improve SCF convergence.

3 Results and Discussions

To obtain Heisenberg exchange constant of molecular magnet, we used DFT+U method and we used U parameter on both the coordinating centers and on the transition metals. The Heisenberg Hamiltonian in general can be written as

$$H = -\sum J_{ij}.S_i.S_J$$

here *J* represents the coupling constant between the two magnetic centers S_i and S_j. The positive *J* values indicate the ferromagnetic ground state and the negative indicate antiferromagnetic ground state.

3.1 Calculation of *J* for Bimettalic Mn(IV) Complexes and Validation of U Values

We started with the X-ray crystal structures of two molecules having bi-manganese (IV) center represented in Fig.1 (complex I) and Fig.2 (complex II). The difference between the two molecules is the acetate bridge in the complex II. According to the previous study,[21] the Hybrid DFT is unable to predict the *J* values for complex II and some other molecules with acetate bridge[21, 22], while it was successful for the complex I. Though BSDFT method is the most used method to predict theoretical *J* values, but for complexes having specific ligand or for very small value of Heisenberg exchange constant, [23-26] this method is not successful so much. That calculation reported the *J* value ~ -37 cm^{-1} whereas the experiment reports -100 cm^{-1}. The reason for failure in predicting *J* suggested as the BS-DFT approach [27] fails to predict the Heisenberg exchange constant related to the delocalization of unpaired electron orbitals over both manganese centers. To deal with this problem we followed

the procedure used by Cao et. al.[20] in order to predict *J*. By adjusting two different *U* values (one for Mn atom, and another for N and O atoms), we obtained a reasonable agreement with experimental *J* values for both complexes I and II. The results are reported in the Table 1. The *J* values for complex I and complex II using BS-DFT method are obtained from reference [21], for complex III [16] the cited *J* is the Heisenberg exchange constant between Mn1 –Mn6′ center.

Table 1. Calculation of Heisenberg exchange constant for Mn complexes

Molecular magnet	Experimental J cm^{-1}	Calculated with BSDFT J cm^{-1}	Calculated with DFT+U J cm^{-1} (this work)	U(Mn), eV	U(O), eV	U(N), eV
Complex I	-147[a]	-131[c]	-177.2	2.5	1.6	1.6
Complex II	-100[b]	-37[c]	-85.9	2.5	1.6	1.6
Complex III Mn1 –Mn6′		0.0[d]	-26.17	2.5	1.6	1.6

[a] Experimental data is taken from [28]
[b] Experimental Data is taken from [29]
[c] BSDFT calculation from ref [21]
[d] BSDFT calculation from ref [16]

Our calculated data agree with the experimental value to within 15%, for both molecule with and without Acetate Bridge, compare to 65% deviation given by broken symmetry Density Functional theory.

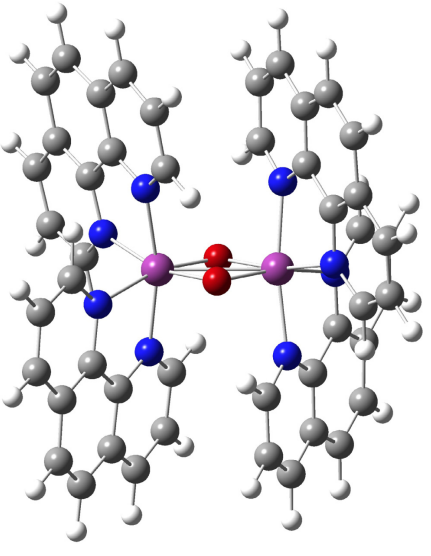


Fig. 1. $[Mn_2O_2(phen)_4]^{4+}$Complex I (violet balls refer to Mn(IV) atoms, grey ones are carbon atoms, white ones are hydrogen, red ones are oxygen, blue ones are nitrogen atoms

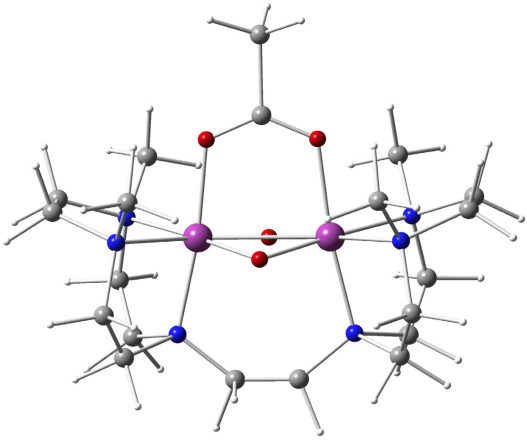


Fig. 2. $[Mn_2O_2(OAc)(Me_4dtne)]^{3+}$ Complex II

3.2 Calculations of *J* for Mn_{12} System

After validating the value for both molecular magnet having Mn(IV) center we applied our protocol for Mn_{12} wheel $[Mn_{12}(O_2CMe)_{14}(mda)_8]$ (where mda is N-methyl diethanolamine). The Mn_{12} wheel has two different valence centers with different coordinations (the Mn(III) is hexa- and Mn(II) is penta-coordinated). Fig. 3 shows the spin arrangement predicted by previous DFT study [2].

The reason for using the parameter *U* for both the p and d orbital was explained by Cao et al.[20] They suggested that Coulomb interactions between oxygen 2*p* electrons are comparable to those between *d* electrons, [30, and 31] and should hence be taken into consideration as well. However, since oxygen usually bares a fully occupied *p*-shell, this correlation effect is often negligible. Therefore, in most cases, DFT+U^{d} can already yield a satisfactory description of the ground state without oxygen 2*p*-electron corrections. Nevertheless, DFT+$U^{d,p}$ has to be taken into consideration explicitly here for both the 3*d* and oxygen 2*p* electrons in order to obtain the correct ground state for this molecule. Previous B3LYP study on this molecule [2] was unable to predict the correct antiferromagnetic ordering for Mn1′- Mn6 and Mn1-Mn6′ center shown in Fig. 4, where zero *J* value was obtained. This study, however, did not consider the entire molecule due to its large size. The molecule was divided into smaller fragments that contained only two or three Mn centers. In our calculation we used all twelve manganese centers and optimized the geometry with PBE exchange-correlation functional. Thus obtained relaxed geometry was then used to calculate the *J* parameter between 6-center fragments, described in Fig. 4. The energy difference between two states shows this center has antiferromagnetic coupling, which is in agreement with the experiment [2] which suggested a *S*=7 ground state of the $[Mn_{12}O_2CMe)(HO_2CMe)_3(OMe)^{2-}(mda)]$.

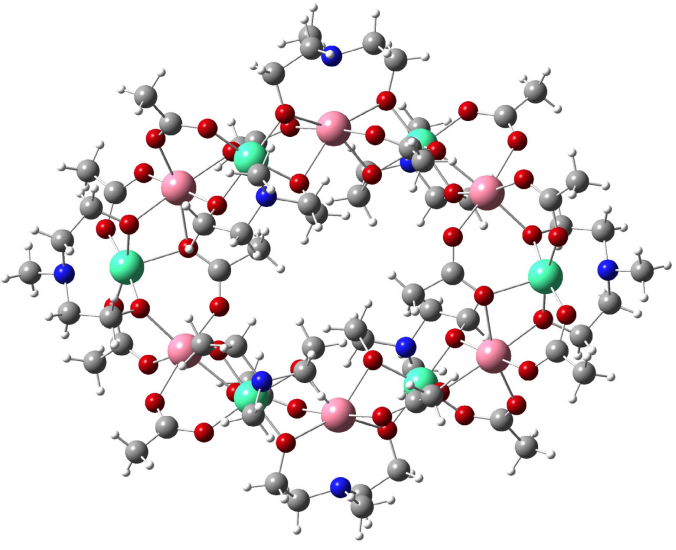


Fig. 3. [MnIIMnIII(O_2CMe)(HO_2CMe)$_3$(OMe)$^{2-}$(mda)] Complex III (pink Mn refers Mn(III) and green ball refers Mn(II)

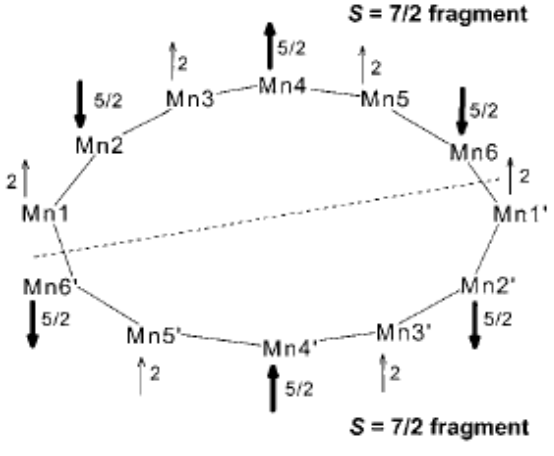


Fig. 4. Depiction of the spin alignments in the *S*=7 ground state [16] of complex III predicted by the DFT calculations, with the Mn1–Mn6′ and Mn1′–Mn6 interactions antiferromagnetic. The dashed line separates the two S=7/2 fragments that are coupled by the interactions between Mn1–Mn6′ and Mn1′–Mn6; if these interactions are antiferromagnetic (negative J values), the resultant spin of the complete molecule is S=7. [16].

4 Conclusions

We have performed DFT and DFT+*U* calculations for two Mn_2 (IV) and one Mn_{12} molecular magnets. Because of the strong magnetic orbital delocalization in these

systems, the broken symmetry pure DFT and hybrid DFT approach fails to predict correct exchange coupling parameter values. The inclusion of a Hubbard-*U* like term for both the Mn 3d and O, N 2p electrons greatly enhances the localization of the magnetic orbitals for both high and low spin states, and is essential in order to obtain the correct ground-state and exchange-coupling parameter values. These properties were successfully reproduced by the Quantum ESPRESSO plane-wave pseudopotential DFT calculations.

Acknowledgements

The authors are grateful to DOE NERSC, I2lab and Institute for Simulation and Training (IST) Stokes HPCC facility at the University of Central Florida for the generous donation of computer time. The authors would like to thank PWscf forum members for useful discussions, Chao Cao for useful comments regarding DFT+U calculations and Zhengji Zhao and Dr. Sampayo Hong for help regarding compiling Quantum Espresso Package in NERSC and I2lab respectively.

References

1. Sessoli, R., Tsai, H.L., Schake, A.R., Wang, S., Vincent, J.B., Folting, K., Gatteschi, D., Christou, G., Hendrickson, D.N.: High-spin molecules: [Mn12O12(O2CR)16(H2O)4]. Journal of the American Chemical Society 115, 1804–1816 (1993)
2. Sessoli, R., Gatteschi, D., Caneschi, A., Novak, M.A.: Magnetic bistability in a metal-ion cluster. Nature (London, United Kingdom) 365, 141–143 (1993)
3. Friedman, J.R., Sarachik, M.P., Tejada, J., Maciejewski, J., Ziolo, R.: Steps in the hysteresis loops of a high-spin molecule. Journal of Applied Physics 79, 6031–6033 (1996)
4. Soler, M., Artus, P., Folting, K., Huffman, J.C., Hendrickson, D.N., Christou, G.: Single-Molecule Magnets: Preparation and Properties of Mixed-Carboxylate Complexes [Mn12O12(O2CR)8(O2CR')8(H2O)4. Inorganic Chemistry 40, 4902–4912 (2001)
5. Soler, M., Wernsdorfer, W., Sun, Z., Ruiz, D., Huffman, J.C., Hendrickson, D.N., Christou, G.: New example of Jahn-Teller isomerism in [Mn12O12(O2CR)16(H2O)4] complexes. Polyhedron 22, 1783–1788 (2003)
6. Rumberger, E.M., Shah, S.J., Beedle, C.C., Zakharov, L.N., Rheingold, A.L., Hendrickson, D.N.: Wheel-Shaped [Mn12] Single-Molecule Magnets. Inorganic Chemistry 44, 2742–2752 (2005)
7. Yang, E.-C., Hendrickson, D.N., Wernsdorfer, W., Nakano, M., Zakharov, L.N., Sommer, R.D., Rheingold, A.L., Ledezma-Gairaud, M., Christou, G.: Cobalt single-molecule magnet. Journal of Applied Physics 91, 7382–7384 (2002)
8. Yoo, J., Yamaguchi, A., Nakano, M., Krzystek, J., Streib, W.E., Brunel, L.C., Ishimoto, H., Christou, G., Hendrickson, D.N.: Mixed-valence tetranuclear manganese single-molecule magnets. Inorganic chemistry 40, 4604–4616 (2001)
9. Aubin, S.M.J., Wemple, M.W., Adams, D.M., Tsai, H.-L., Christou, G., Hendrickson, D.N.: Distorted MnIVMnIII3 Cubane Complexes as Single-Molecule Magnets. Journal of the American Chemical Society 118, 7746–7754 (1996)
10. Yoo, J., Wernsdorfer, W., Yang, E.-C., Nakano, M., Rheingold, A.L., Hendrickson, D.N.: One-Dimensional Chain of Tetranuclear Manganese Single-Molecule Magnets. Inorganic Chemistry 44, 3377–3379 (2005)

11. Beedle, C.C., Heroux, K.J., Nakano, M., DiPasquale, A.G., Rheingold, A.L., Hendrickson, D.N.: Antiferromagnetic tetranuclear manganese complex: Wheel or dicubane? Polyhedron 26, 2200–2206 (2007)
12. Scott, R.T.W., Milios, C.J., Vinslava, A., Lifford, D., Parsons, S., Wernsdorfer, W., Christou, G., Brechin, E.K.: Making 'wheels' and 'cubes' from triangles. Dalton Transactions, 3161–3163 (2006)
13. Affronte, M., Casson, I., Evangelisti, M., Candini, A., Carretta, S., Muryn, C.A., Teat, S.J., Timco, G.A., Wernsdorfer, W., Winpenny, R.E.P.: Linking rings through diamines and clusters: Exploring synthetic methods for making magnetic quantum gates. Angewandte Chemie, International Edition 44, 6496–6500 (2005)
14. Murugesu, M., Wernsdorfer, W., Abboud, K.A., Christou, G.: New structural motifs in manganese single-molecule magnetism from the use of triethanolamine ligands. Angewandte Chemie, International Edition 44, 892–896 (2005)
15. Manoli, M., Prescimone, A., Mishra, A., Parsons, S., Christou, G., Brechin, E.K.: A high-spin molecular wheel from self-assembled 'Mn rods'. Dalton Transactions, 532–534 (2007)
16. Foguet-Albiol, D., O'Brien, T.A., Wernsdorfer, W., Moulton, B., Zaworotko, M.J., Abboud, K.A., Christou, G.: DFT computational rationalization of an unusual spin ground state in an Mn12 single-molecule magnet with a low-symmetry loop structure. Angewandte Chemie, International Edition 44, 897–901 (2005)
17. Baroni, S.e.a.: Quantum-ESPRESSO (2006), http://www.pwscf.org
18. Vanderbilt, D.: Soft Self-Consistent Pseudopotentials in a Generalized Eigenvalue Formalism. Physical Review B 41, 7892–7895 (1990)
19. Marzari, N., Vanderbilt, D., Payne, M.C.: Ensemble density-functional theory for ab initio molecular dynamics of metals and finite-temperature insulators. Physical review letters 79, 1337–1340 (1997)
20. Cao, C., Hill, S., Cheng, H.-P.: Strongly Correlated Electrons in the [Ni(hmp)(ROH)X]4 Single Molecule Magnet: A DFT+U Study. Physical Review Letters 100, 167206/167201–167206/167204 (2008)
21. Rudberg, E., Salek, P., Rinkevicius, Z., Agren, H.: Heisenberg exchange in dinuclear manganese complexes: A density functional theory study. Journal of Chemical Theory and Computation 2, 981–989 (2006)
22. Zhao, X.G., Richardson, W.H., Chen, J.L., Li, J., Noodleman, L., Tsai, H.L., Hendrickson, D.N.: Density functional calculations of electronic structure, charge distribution, and spin coupling in manganese-oxo dimer complexes. Inorganic chemistry 36, 1198–1217 (1997)
23. Taylor, P.R.: Weakly coupled transition-metal centres: High-level calculations on a model Fe(IV)-Fe(IV) system. Journal of Inorganic Biochemistry 100, 780–785 (2006)
24. Ghosh, A., Taylor, P.R.: High-level ab initio calculations on the energetics of low-lying spin states of biologically relevant transition metal complexes: first progress report. Curr. Opin. Chem. Biol. 7, 113–124 (2003)
25. Ciofini, I., Daul, C.A.: DFT calculations of molecular magnetic properties of coordination compounds. Coordination Chemistry Reviews 238, 187–209 (2003)
26. Ali, M.E., Datta, S.N.: Theoretical investigation of magnetic properties of a dinuclear copper complex [Cu-2(mu-OAc)(4)(MeNHph)(2)]. Journal of Molecular Structure-Theochem. 775, 19–27 (2006)
27. Rudberg, E., Salek, P., Rinkevicius, Z., Agren, H.: Heisenberg Exchange in Dinuclear Manganese Complexes: A Density Functional Theory Study. J. Chem. Theory Comput. 2, 981–989 (2006)

28. Stebler, M., Ludi, A., Burgi, H.B.: [(Phen)2Mn-4(m-O)2Mn-3(Phen)2](PF6)3.CH3CN and [(Phen)2Mn4(m-O)2Mn4(Phen)2](ClO4)4CH3CN (Phen = 1,10-Phenantholine) - crystal structure analyses at 1000K, interpretation of disorder, and optical, magnetic and electrochemical results. Inorganic chemistry 25, 4743–4750 (1986)
29. Schafer, K.O., Bittl, R., Zweygart, W., Lendzian, F., Haselhorst, G., Weyhermuller, T., Wieghardt, K., Lubitz, W.: Electronic structure of antiferromagnetically coupled dinuclear manganese ((MnMnIV)-Mn-III) complexes studied by magnetic resonance techniques. Journal of the American Chemical Society 120, 13104–13120 (1998)
30. McMahan, A.K., Martin, R.M., Satpathy, S.: Calculated effective Hamiltonian for lanthanum copper oxide (La2CuO4) and solution in the impurity Anderson approximation. Physical Review B: Condensed Matter 38, 6650–6666 (1988)
31. Yoo, J., Brechin, E.K., Yamaguchi, A., Nakano, M., Huffman, J.C., Maniero, A.L., Brunel, L.C., Awaga, K., Ishimoto, H., Christou, G., Hendrickson, D.N.: Single-molecule magnets: a new class of tetranuclear manganese magnets. Inorganic chemistry 39, 3615–3623 (1988)

A Cheminformatics Approach for Zeolite Framework Determination

Shujiang Yang[1], Mohammed Lach-hab[1], Iosif I. Vaisman[1,2], and Estela Blaisten-Barojas[1,3]

[1] Computational Materials Science Center, George Mason University, MSN 6A2, Fairfax, Virginia 22030, USA
[2] Department of Computational Biology and Bioinformatics, George Mason University, MSN 5B3, Manassas, Virginia 20110, USA
[3] Department of Computational and Data Sciences, George Mason University, MSN 6A2, Fairfax, Virginia 22030, USA
blaisten@gmu.edu

Abstract. Knowledge of the framework topology of zeolites is essential for multiple applications. Framework type determination relying on the combined information of coordination sequences and vertex symbols is appropriate for crystals with no defects. In this work we present an alternative machine learning model to classify zeolite crystals according to their framework types. The model is based on an eighteen-dimensional feature vector generated from the crystallographic data of zeolite crystals that contains topological, physical-chemical and statistical descriptors. Trained with sufficient known data, this model predicts the framework types of unknown zeolite crystals within 1-2 % error and shows to be better suited when dealing with real zeolite crystals, all of which always have geometrical defects even when the structure is resolved by crystallography.

1 Introduction

Zeolites are crystalline materials with regular structures consisting of molecular-sized pores and channels. These crystals are widely used in the field of adsorption, ion-exchange, heterogeneous catalysis (basically all gasoline production employs zeolite catalysts), as well as in health applications, sensors, solar energy conversion [1]. There are hundreds of zeolite species occurring naturally and/or synthetically, and millions more have been hypothetically proposed [2]. Zeolite crystals are constructed from an underlying three-dimensional network of TO_4 building block units. Within this network there are loosely bonded exchangeable cations, adsorbent phases and the building block central atom is tetrahedrally coordinated with four oxygen atoms. Predominantly, Si, Al or P is the element in the center of the tetrahedral building blocks and is referred as the T-atom. Zeolite networks are constructed by spatially accommodating the TO_4 building blocks by corner sharing the oxygens located at their vertices. These networks span a certain length and then repeat periodically along the crystal. Thus, these underlying networks depend directly on the connectivity of TO_4 units. Other zeolite crystal components such as cations, adsorbent phase,

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 160–168, 2009.

chemical composition, and observed crystallography properties are irrelevant in the determination of the underlying network. There are topological differences between networks in different crystals. Once a crystal network possesses a recognized topology, the Structure Commission of the International Zeolite Association (IZA-SC) approves it as an established framework type [3] and crystals displaying one of the approved framework types are then cataloged as zeolites. Non-approved network topologies fall into the category of *hypothetical* zeolites or *are not* zeolites. The IZA-SC currently recognizes 186 unique framework topologies [3]. The IUPAC Commission on Zeolite Nomenclature assigns a three-capital-letter acronym, the framework type codes (FTC), to each framework topology [4]. Known zeolite crystals belong to one of these topological categories. Crystals suspected to be zeolites that do not meet the established framework types cannot be cataloged as zeolites.

The FTC is conventionally determined with the combined information of the coordination sequences (CS) [5] and vertex symbols (VS) [6] of a zeolite crystal. Although it is not excluded that different framework types would have identical coordination sequences and vertex symbols, these cases are infrequent [7]. The conventional CS-VS method has limitations when applied to real zeolite crystals. Indeed, we noticed that multiple zeolite crystals in the Inorganic Crystal Structure Database (ICSD) [8] are distorted in various ways or might not contain complete crystal information. Coordination sequences or vertex symbols calculated for these crystals are erroneous and as a consequence their FTC cannot be predicted.

Exploitation of data mining and machine learning approaches is emerging in recent years in the field of chemical and materials informatics as a powerful approach for designing models that are developed based on data archived in databases [9,10]. The challenging task is to turn such models uniquely based on data analysis into novel applications. Similar approaches have been successful in a diversity of fields ranging from speech and vision recognition, robot control, business management, to bioinformatics and drug design.

In this work, we introduce a machine learning methodology for classifying zeolite crystals according to their framework type. The model presented here is the second-generation Zeolite-Structure-Predictor (ZSP2) developed on a data set of around 1300 zeolite crystals contained in the ICSD. ZSP2 is an extension of the original ZSP model [10] in which different topological descriptors are considered. The methodology used for building this model can be easily ported for the structural analysis of other families of crystals.

2 Methodology

The ZSP2 uses Breiman's Random Forest (RF) algorithm [11], which consists of an ensemble of decision trees trained on a bootstrap sample of the training data. The algorithm considers random groups of attributes for creating many trees rather than using all attributes to build one tree. Classification predictions are made by majority vote of all the trees. In this work the forest contains 100 trees and the WEKA [12] implementation of RF is used throughout.

2.1 Data Preprocessing

The process of cleaning the data is of paramount importance in data mining. When queried for zeolite crystals, the ICSD gives about 1600 crystal entries. Data in these entries are collected from published literature and do not include the framework type information. Based on the structure content in each crystal entry, we were able to assign the CS and VS to 1473 crystal entries by means of the zeoTsites package [13]. The remaining crystal entries have spurious geometry disorder or insufficient information in the database and no CS-VS could be determined precisely [14]. The CS and VS of these 1473 crystals were compared with the IZA-SC table, confirming that 1370 crystals can be referred as zeolites belonging to 94 framework types. The conventional CS-VS method proves incapable for identifying a framework type of the remaining 103 entries. Therefore the CS-VS method fails in 7.5 % of the cases to assign a framework type to a suspected zeolite crystal.

In machine learning terminology a framework type is referred as a *class* and each zeolite is referred as an *instance*. Machine learning models are more robust when classes are populated by a large number of instances. The 1370 zeolite entries are unevenly distributed among the 94 framework types. Indeed, class population ranges from 1 to 351 instances. There are 53 classes populated with only one or two instances, which are clearly inadequate for developing a data-based model. Because of this limitation, our machine learning study focuses on the 41 classes populated with at least 3 instances. However, the model described in this work can be easily extended to classify according to 186 classes in the zeolite case, and can be used for other crystal families as well.

Although the ICSD is the largest and most comprehensive database of inorganic crystals, it presents constrains for building models based on data contained in its repository. There is hope that the ICSD will continue adding crystals to the existing portfolio, which would then allow for further informatics approaches based on the crystal data to become useful to the materials and solid state chemistry community.

2.2 Feature Generation

An *attribute* is a descriptor of a certain crystal property. A *feature* is the specification of an attribute. The ZSP2 model for classification of zeolites into framework types includes categorical and quantitative features of topological, chemical and statistical nature.

The topological descriptors in the ZSP2 are based on a statistical geometry approach based on the Delaunay tessellation [15] of a supercell of each zeolite crystal [10]. Delaunay tessellation provides an objective, non-arbitrary definition of nearest neighboring points in space. Depending on the motif of the points, such tessellation has been used to characterize liquids [16,17], proteins [18], as well as zeolites [19]. The ICSD crystal entries provide the asymmetric unit cell of the crystal resolved from X-ray experiments. With this information, it is possible to generate the unit cell of a given crystal and once the unit cell is known, a supercell containing several unit cells can be generated numerically [20]. In the zeolites analyzed, the unit cells span a wide range of sizes and contain between 20 and 3040 atoms (excluding the hydrogens).

With the purpose of proposing topological descriptors, large supercells of all 1370 zeolites are generated such that a spherical cut of fixed radius 35.32 Å could be carved out of each supercell. The sphere radius is chosen to ensure that the carved sphere in crystals with huge unit cells encompasses a central unit cell and at least one neighboring unit cell in each of the three directions. The next step is to remove from the supercell sphere all oxygen atoms, all cations, and the full adsorbent phase. Thus, only T-atoms are retained inside the sphere. These T-atoms constitute the backbone of the zeolite framework and the Delaunay tessellation is performed on points in space coinciding with their location. This procedure yields the Delaunay simplices (distorted tetrahedra), which contain T-atoms at their vertices. Tens of thousands of Delaunay simplices are obtained per zeolite spherical supercell. Most simplices are distorted tetrahedra with edges that can be very long. In contrast, the TO_4 units that sustain the zeolite framework are near-to-perfect tetrahedra with edge lengths consistent with small variations around the oxygen-oxygen bond length.

In this work, the proposed topological descriptors are based on three geometrical properties of the Delaunay simplices: i) mean edge length ($\overline{d}$) of the six edges of each simplex; ii) in-sphere volume (*iV*) of the largest sphere inscribed in a simplex; iii) tetrahedrality (*T*) defined as the degree of distortion of a simplex from a regular tetrahedron:

$$T = \sum_{i=1}^{5} \sum_{j=i+1}^{6} \frac{(d_i - d_j)^2}{15\overline{d}^2} ,$$

where d_i is the *i*-th edge length of the simplex. Mean and standard deviation (σ) of these three properties are calculated for all simplices within each zeolite. The six topological descriptors are *mean_*$\overline{d}$, σ_$\overline{d}$, *mean_iV*, σ_*iV*, *mean_T*, and σ_*T*.

Additional geometrical descriptors were adopted by considering secondary simplices corresponding to a second coordination shell in Delaunay space [21]. Because each simplex has four adjacent tetrahedra that share one of its faces, the four new vertices can be linked into a larger tetrahedron defining the secondary simplex. Six geometrical descriptors: *mean_*$\overline{d}_2$, σ_$\overline{d}_2$, *mean_iV*$_2$, σ_*iV*$_2$, *mean_T*$_2$, and σ_*T*$_2$, based on secondary simplices were adopted.

Finally, six physical and chemical properties of a crystal are considered as descriptors: framework density (ρ), unit cell volume (V_o), space group (*SG*), and the chemical composition of T-atoms Si, Al and P (*[Si], [Al], [P]*). Among them, *SG* is the only nominal feature.

In summary, the zeolite classifier model ZSP2 is based on an 18-feature vector composed of twelve topological/statistical descriptors and six physical-chemical descriptors.

3 Results and Discussion

The performance of the ZSP2 model depends on the size of the feature vector used to create it. The performance is measured in terms of *accuracy*, which is defined as the percentage of instances that the model classifies correctly. Traditional classifiers in data mining contain very few classes, and typically the classification is reduced to two classes, which can then be addressed in binary language. Considering the dataset of

1370 instances and 41 classes, classification accuracy increases progressively as additional relevant features increase the dimensionality of the feature vector. This effect is shown in Table 1, where the reported accuracy is calculated with stratified 10-fold cross validation and averaged over ten trials. Classification accuracy is 90.6% with a feature set containing the six topological descriptors based on the first Delaunay shell (*mean_*$\bar{d}$, σ_$\bar{d}$, *mean_iV*, σ_*iV*, *mean_T*, and σ_*T*). By gradually adding features from secondary simplices, T-atom composition, ρ, *Vo*, and *SG*, the classification performance is progressively improved. Finally, with the 18 features included in the model an impressive accuracy of 97.9% is reached. Consistently, the out of bag error (OOB) decreases as classification accuracy increases.

The worth of features for this classifier was investigated by evaluating their information gain with respect to the classes considered. This analysis determines that *mean* _$\bar{d}$, *mean_iV*, *mean_T* , ρ, V_o, *SG, [Si], [Al]* are the nine most significant features. However, the ZSP2 built with the18-feature set is computationally very fast and thus all 18 features are kept throughout this study.

To analyze the effect of the population size of each class on the ZSP2, seven different models were built each of them classifies into a similar number of classes, but these classes are populated with different number of instances per class (x). Figure 1 illustrates the performance of the seven classifications. The plotted accuracy is a result of using stratified 10-fold cross validation and averaging over ten trials. Although the classification process is dependent on the motif of classes involved in each data group, it is evident that the ZSP2 model is more accurate when trained with well populated classes. In fact, the ZSP2 yields 100% accuracy when built with six classes that have more than 63 instances.

Table 1. ZSP2 classification with various feature sets, 1370 instances and 41 classes

Feature set	*mean_*$\bar{d}$, σ_$\bar{d}$, *mean_iV*, σ_*iV*, *mean_T*, σ_*T*	*mean_*$\bar{d}$, σ_$\bar{d}$, *mean_iV*, σ_*iV*, *mean_T*, σ_*T*, *mean_*$\bar{d}$ *2*,σ_$\bar{d}$ *2*, *mean_iV2*,σ_*iV2*, *mean_T2*, σ_*T2*	*mean_*$\bar{d}$, σ_$\bar{d}$, *mean_iV*, σ_*iV*, *mean_T*, σ_*T*, *mean_*$\bar{d}$ *2*,σ_$\bar{d}$ *2*, *mean_iV2*,σ_*iV2*, *mean_T2*, σ_*T2* *[Si], [Al], [P]*	*mean_*$\bar{d}$, σ_$\bar{d}$, *mean_iV*, σ_*iV*, *mean_T*, σ_*T*, *mean_*$\bar{d}$ *2*,σ_$\bar{d}$ *2*, *mean_iV2*,σ_*iV2*, *mean_T2*, σ_*T2* *[Si], [Al], [P],* ρ, V_o, *SG*
OOB	0.024±0.001	0.099±0.004	0.054±0.003	0.024±0.001
Accuracy (%)	90.6±0.2	92.8±0.2	94.6±0.3	97.9±0.1

The model improvement through experience is demonstrated through its learning curve. For this experiment, a balanced dataset is constructed such that each class is populated equally. Figure 2 shows the learning curve of the ZSP2 when the model classifies into six classes, each of them populated with 60 instances. In this figure the accuracy (vertical axis) pertains to instances correctly classified from split of the total number of instances (*test*-set) once the model was *trained* with the remaining split of

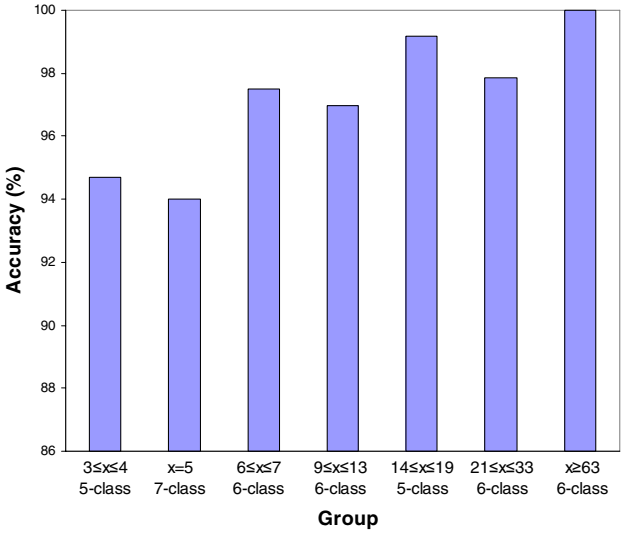


Fig. 1. Classification performance of the ZSP2 obtained for seven data groupings with about constant number of classes. "*x*" is number of instances per class.

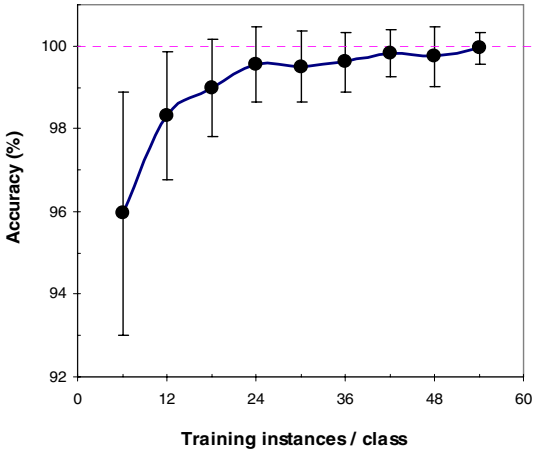


Fig. 2. The learning curve of the ZSP2 built for a balanced dataset of 360 instances and six classes

available instances (plotted on the abscissa). For each training/test split, the training instances are drawn at random from the available data. Next, the test instances are randomly drawn from the remaining instances. The split is repeated 100 times for each point. Both mean and standard deviation of this process are shown in Figure 2. It is clearly shown that the ZSP2 improves fast when the training set is small, then more smoothly when the training set exceeds 24 instances/class, and reaches a plateau at

about 42 instances/class. Finally, the ZSP2 model performs perfectly when trained with 54 instances per class. Therefore, the ZSP2 yields perfect classification into six classes for datasets containing 60 instances per class.

Among the 1370 available instances, 1041 of them are distributed in 11 classes with more than 27 instances/class. The ZSP2 model for this 11-class dataset, using stratified 10-fold cross validation and repeated ten times, classifies 99.3% of the instances correctly, which is not perfect but excellent.

The ZSP2 could be tested with larger datasets and more than 41 classes. However, we have been limited to the content of the ICSD, which currently allows for as much as 41-class classification as shown in Table 1. To predict the classification of instances falling within a class not included in the 41 trained classes, a *bag* class was defined containing the 53 frameworks poorly populated in the ICSD. Now ZSP2 is built with 1370 instances to classify into 42 classes including the bag-class. With ten times 10-fold cross validation, the ZSP2 correctly classifies 95.3% of all instances. If the number of classes is reduced by keeping in the ZSP2 only those classes populated with x or more instances, and placing the rest into the bag-class, the predictive power of the model improves as the number of classes decreases as shown in Table 2.

Table 2. The ZSP2 classification of four datasets including a bag-class

Dataset	Dataset1	Dataset2	Dataset3	Dataset4
Number of instances per class=x	x≥3	x≥9	x≥19	x≥63
Size of the bag class	65	157	251	477
OOB	0.052±0.003	0.037±0.003	0.027±0.002	0.009±0.001
Accuracy (%)	95.3±0.1	96.5±0.1	97.3±0.1	99.0±0.2

During the data cleaning procedure 103 instances with determined CS-VS were removed from the analysis because they were not consistent with any framework type. The ZSP2 for a 41-class model predicts that 60 of these instances belong to these 41 classes. This finding was further corroborated by details given in the published literature that originated the entries in the ICSD of these 60 crystals.

Compared with the conventional method to assign zeolite framework types, the ZSP2 machine learning model is more robust to geometry disorder and occasional errors in the data. By examining the zeolite entries in the ICSD, it was noticed that measured structural crystal data may differ substantially from perfect crystals. As a consequence, erroneous FTC would be determined for these cases. On the other hand, these types of errors are more likely to be tolerated by the ZSP2 model.

4 Conclusion

In this work we present the ZSP2, a machine learning model for classifying zeolites crystals according to their framework type. The approach requires as input the resolved crystallographic data of each crystal only for T-atoms in the framework. The ZSP2 is then a more efficient model than the ZSP where the crystallographic resolution had to contain all atoms. The complete crystallographic information is also required to assign a framework type to a zeolite crystal using the conventional coordination sequence and vertex symbol approach. The ZSP2 performance is highly accurate. Indeed, the model is able to predict correct classification with up to 100% accuracy when enough data are available. The novel approach is considerably more robust than the conventional identification method and can potentially be used to study other families of crystals. There are over 100,000 crystal entries in the ICSD and the ZSP2 model can be tailored for clustering and classifying with a variety of different objectives. Work is in progress in this direction.

Acknowledgments. This work was supported by the National Science Foundation grant CHE-0626111. Authors acknowledge the NIST Standard Reference Data Program for making available the ICSD zeolite data set.

References

1. Payra, P., Dutta, P.K.: Zeolites: a Primer. In: Auerbach, S.M., Carrado, K.A., Dutta, P.K. (eds.) Handbook of Zeolite Science and Technology, pp. 1–19. Marcell Dekker, New York (2003)
2. Foster, M.D., Treacy, M.M.J.: A Database of Hypothetical Zeolite Structures, http://www.hypotheticalzeolites.net
3. IZA-SC and its Standard Database of Zeolite Frameworks, http://www.iza-structure.org/databases/
4. Barrer, R.M.: Chemical Nomenclature and Formulation of Compositions of Synthetic and Natural Zeolites. Pure Appl. Chem. 51, 1091–1100 (1979)
5. Meier, W.M., Moeck, H.J.: The Topology of Three-dimensional 4-Connected Nets: Classification of Zeolite Framework Types Using Coordination Sequences. J. Solid State Chem. 27, 349–355 (1979)
6. O'Keeffe, M., Hyde, S.T.: Vertex Symbols for Zeolite Nets. Zeolites 19, 370–374 (1997)
7. Treacy, M.M.J., Foster, M.D., Randall, K.H.: An Efficient Method for Determining Zeolite Vertex Symbols. Micropor. Mesopor. Mater. 87, 255–260 (2006)
8. Inorganic Crystal Structure Database (ICSD), http://www.nist.gov/srd/nist84.htm
9. Fischer, C.C., Tibbetts, K.J., Morgan, D., Ceder, G.: Predicting Crystal Structure by Merging Data Mining with Quantum Mechanics. Nature Mater. 5, 641–646 (2006)
10. Carr, D.A., Lach-hab, M., Yang, S., Vaisman, I.I., Blaisten-Barojas, E.: Machine Learning Approach for Structure-based Zeolite Classification. Micorpor. Mesopor. Mater. 117, 339–349 (2009)
11. Breiman, L.: Random Forests. Machine Learning 45, 5–32 (2001)
12. Weka 3: Data Mining Software in Java, http://www.cs.waikato.ac.nz/ml/weka/

13. Sastre, G., Gale, J.D.: Zeotsites: A Code for Topological and Crystallographic Tetrahedral Sites Analysis in Zeolites and Zeotypes. Micorpor. Mesopor. Mater. 43, 27–40 (2001)
14. Yang, S., Blaisten-Barojas, E.: Internal communication to the ICSD
15. Delaunay, B.N.: Sur La Sphere Vide. Bull. Acad. Sci. USSR (in Russian) 7, 793–800 (1934)
16. Medvedev, N.N., Naberukhin, Y.I.: Analysis of Structure of Simple Liquids and Amorphous Solids by Method of Statistical Geometry. Zh. Strukt. Khimii 28, 117–132 (1987)
17. Vaisman, I.I., Brown, F.K., Tropsha, A.: Distance Dependence of Water Structure around Model Solutes. J. Phys. Chem. 98, 5559–5564 (1994)
18. Vaisman, I.I.: Statistical and Computational Geometry of Biomolecular Structure. In: Gentle, J.E., Härdle, W., Mori, Y. (eds.) Handbook of Computational Statistics, pp. 981–1000. Springer, New York (2004)
19. Foster, M.D., Rivin, I., Treacy, M.M.J., Friedrichs, O.D.: A Geometric Solution to the Largest-free-sphere Problem in Zeolite Frameworks. Micropor. Mesopor. Mater. 90, 32–38 (2006)
20. Computational Crystallography Toolbox, `http://cctbx.sourceforge.net/`
21. Wako, H., Yamato, T.: Novel Method to Detect a Motif of Local Structures in Different Protein Conformations. Protein Engineering 11, 981–990 (1998)

Theoretical Photochemistry of the Photochromic Molecules Based on Density Functional Theory Methods

Ivan A. Mikhailov[1] and Artëm E. Masunov[1,2,3]

[1] NanoScience Technology Center
[2] Department of Chemistry
[3] Department of Physics, University of Central Florida, 12424 Research Parkway, Suite 400, Orlando, FL 32826, USA
amasunov@mail.ucf.edu

Abstract. Mechanism of photoswitching in diarylethenes involves the light-initiated symmetry-allowed disrotatory electrocyclic reaction. Here we propose a computationally inexpensive Density Functional Theory (DFT) based method that is able to produce accurate potential surfaces for the excited states. The method includes constrained optimization of the geometry for the ground and two excited singlet states along the ring-closing reaction coordinate using the Slater Transition State method, followed by single-point energy evaluation. The ground state energy is calculated with the broken-symmetry unrestricted Kohn-Sham formalism (UDFT). The first excited state energy is obtained by adding the UDFT ground state energy to the excitation energy of the pure singlet obtained in the linear response Time-Dependent (TD) DFT restricted Kohn-Sham formalism. The excitation energy of the double excited state is calculated using a recently proposed (Mikhailov, I. A.; Tafur, S.; Masunov, A. E. Phys. Rev. A 77, 012510, 2008) *a posteriori* Tamm-Dancoff approximation to the second order response TD-DFT.

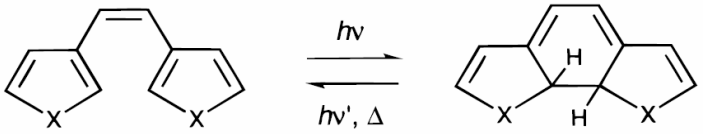
Keywords: Time Dependent Density Functional Theory, Photochromism, Photoswitching, Optical Data Storage, Double excited state, Theoretical Photochemistry, Two-photon Absorption, Rational Materials design.

1 Introduction

Recording density of the data storage becomes an important issue in the recent years. While magnetic media neared its maximum capacity with the bit size c.a. 20 nm, the technological advances in optical disks is expected to win the competition with traditional magnetic storage devices. In particular, these advances include the photon-mode recording, in the contrast to the optical memory systems presently available on the market. Most of the existing systems utilize heat-mode recording, where the light is converted into thermal energy, induces a magnetic or structural phase transition (magneto-optical [1, 2] and phase-change [3] effects) and changes physical properties of the medium. In the photon-mode of data recording, the light initiates photochemical reaction of a particular component of the material. This allows introducing the third axial dimension to the recording process. This three-dimensional technology will use

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 169–178, 2009.

polymers, doped with the photochromic compounds undergoing a reversible photoisomerization [4]. A promising class of photochromics is exemplified by diarylethene compounds, shown in Scheme 1. They undergo photoinduced conrotatory ring opening and closing, and have important practical advantages over other classes of compounds, including thermal stability and resistance to linear optical photofatigue [5].



Scheme 1. Diarylethene photoswitching reaction (X = O or S)

Some of the photochromic materials undergo photoisomerization in ultrafast regime (in the order of 10 fs). These ultrafast switching capabilities can be useful for various photonic devices, such as optical switches, variable frequency filters, attenuators, and phase shifters, interconnection, and components of optical computers.

Organic photonic materials have another important advantage, as their properties can be fine-tuned by chemical modifications of molecules. However, these modifications may change or completely eliminate the photochromic ability. Theoretical studies are indispensible to understand the reason of these changes and to formulate the guiding principles for the rational molecular design ([6] and [7]). However, as Nakamura *et al*. state in their recent review [7], the accurate relative energies of the excited states in real size molecules are still very difficult to calculate, because it requires the balanced description for both covalent (2A state) and ionic states (1B state).

In this contribution we propose a new theoretical method based on the Density Functional Theory. This method is able to produce accurate potential surfaces for the 1B and 2A excited states as compared to available experimental data and results of the high-level multireference wavefunction theory methods. It is also computationally inexpensive and capable to predict the photophysics of large molecules of practical interest.

2 Theory

Computational photochemistry offers a number of theoretical methods for investigation of photochemical reaction mechanisms. Unlike thermally activated chemical reactions, which take place in the ground electronic state (S_0), a photochemical process involves the electronically excited state (S_1). During this process the reactive system is electronically excited from S_0 to S_1, and after some evolution on an upper potential energy surface (PES) decays back to the ground state in either product or reactant basin through conical intersections (CIX). A useful simplified description of this process is called the reaction Pathway Approach [8]. Instead of the entire PES, it considers the minimum energy path (MEP) [9] which is followed by the center of a wave packet [10]. This approach is focused on local properties of PES, such as minima, barriers, and slopes. In the Pathway Approach, a Conical Intersection serves as a funnel, which

delivers the excited state intermediate to the ground state reactant or the product, so that quantum yield is largely determined there [11]. It has been found very useful for a qualitative analysis of reaction mechanisms, prediction of photoproducts, and rationalization of experimental excited state lifetimes, quantum yields, absorption and emission spectra [12, 13].

A number of computational tools have been developed to predict the PES [11, 14, 15]. A computationally inexpensive approach to describe electronically excited systems is based on the time-dependent (TD) or, more precisely, linear response DFT formalism. Instead of orbital relaxation, TD-DFT uses a mathematically equivalent procedure where the KS wavefunction is expanded in terms of Slater determinants, singly excited with respect to the reference state. The rigorous formulation of TD-DFT [16] demonstrates that this description is in principle exact, given that the frequency-dependent exchange-correlation functional is known. In most practical applications, however, this frequency dependence is ignored (so called adiabatic TD-DFT). This method was often reported to accurately predict electronic spectra and excited state geometries. However, TD-DFT was found to be somewhat less successful in description of PESs in the vicinity of a CIX [17]. In this contribution we show that these difficulties are routed to the failure of the restricted Kohn-Sham formalism for the reference ground state close to geometry of the pericyclic minimum, and introduce a possible solution.

The Kohn-Sham formalism of DFT was developed for non-degenerate cases; it breaks down for systems with strong diradical character and degeneracy of the electronic levels, such as CIX geometries. However, static (also known as left-right) electron correlation can be taken into account by using different orbitals for different spin. This approach, known as the unrestricted Kohn-Sham formalism (UKS) is known to yield a qualitatively correct description of bond breaking [18].

Excited states, on the other hand, require the restricted formalism to avoid heavy mixing of higher spin states in description of the excited singlet. Although the TD-DFT was suggested on the UKS reference [19], this is considered to be incorrect in a rigorous theory [20]. One possible approach for analyzing PES of excited states can be formulated by adding excitation energies obtained in the restricted TD-DFT formalism to the ground state energies calculated with the UKS method. Thereafter we will refer to this approach as to RTD-UDFT. Although for the photoswitching systems considered in this contribution the difference in the ground state energy obtained with the RKS and UKS formalisms is close to 20 kcal/mol or less, we will numerically show that this difference is sufficient to bring the excited state PESs to agreement with the results obtained at a higher theory level, when available.

Another theoretical development, necessary to describe the region of the conical intersection is related to the double excited states, missing in the adiabatic linear response TD-DFT approximation [21]. Mixing of the double excited states to the linear response TD-DFT states is offered by the Coupled Electronic Oscillator formalism [22-24], where doubly excited states appear in the second order as simple products of the excitations obtained at the linear response level. We recently used this fact to propose the *a posteriori* Tamm-Dancoff approximation (ATDA), and demonstrated its accuracy for linear polyenes in their ground state geometry [25]. We will show in Section 5, that ATDA-UDFT produces accurate energies for the double excited state in the entire range of the bond breaking reaction coordinate, provided that the molecular

geometry corresponds to that state. For excited states that appear in the linear response TD-DFT the ATDA yields identical excitation energies and transition dipoles from the ground state, while permanent dipoles and state-to-state transition dipoles differ.

Since analytical gradients in the ATDA-UDFT approach are not yet implemented in computer codes, in our studies we use the Slater transition state method (STS) to optimize geometry of the excited states. In this method, half an electron is promoted from the highest occupied molecular orbital to the lowest unoccupied molecular orbital and self-consistency is achieved with these fractional orbital occupations [26]. STS is known to be a good approximation to the corresponding ΔSCF excitation energy [27, 28]. Its further extension to the modified linear response DFT method [29] yields considerable improvement in description of the charge-transfer and Rydberg states, compared to the TD-DFT approach. A practical advantage of STS is an easier SCF convergence, compared to the excited-state SCF convergence, which often presents a major problem [30].

3 Computational Details

All calculations were performed using the Gaussian 2003 Rev. E1 suite of programs [31]. We used the hybrid meta-GGA exchange-correlation functional M05-2X from Truhlar's group with double fraction of the Hartree-Fock exchange, designed for accurate description of both equilibrium geometries and transition states. The minimum energy pathways (MEPs) were built using the relaxed scan along the forming pericyclic C–C bond (reaction coordinate). The ground singlet state (S0), was optimized at the UM05-2X/6-31G level of theory, while the single 1B and double 2A excited singlet state geometries were optimized in the Slater Transition State method. STS was implemented using equal fractional occupation numbers for HOMO and LUMO in the alpha-set only [to approximate geometry of the single excited state 1B, IOp(5/75=1,76=2)], and both alpha and beta sets [to approximate geometry of the double excited state 2A, IOp(5/75=1, 76=2,77=1,78=2)]. Excitation energies were taken from single-point calculations in the *a posteriori* Tamm-Dancoff approximation for the lowest single excited state 1B and the lowest double excited state 2A. The excitation energies thus obtained at the ATDA-M05-2X/6-31 level, were added to the ground state energies obtained at the UM05-2X/6-31G level of theory. The resulting ATDA-UM05-2X/6-31G//STS-UM05-2X/6-31G energies were plotted in the range of the reaction coordinate from 1.4 to 3.5A with 0.1A step size as MEPs.

4 Results and Discussion

Cyclohexadiene/hexatriene (CHD/HT) conversion is the simplest example of an electrocyclic reaction. Dynamics of cycloreversion in the CHD/HT system was repeatedly studied with time-resolved ultrafast spectroscopy techniques [12, 32]. The results are summarized in Ref. [10].

The theoretical description of this process involves plotting realistic potential energy surfaces (PESs), which until recently required the use of *ab initio* multireference-based quantum chemistry methods. Pioneering CAS study of the HT/CHD system

was published by Robb, Olivucci *et al*. [33-35]. They found that 2A and 1A surfaces touch at a molecular conformation of tetraradical character, located away from the C_2-symmetric coordinate, and including partial bonds C1…C6 and C2…C6. They also found that accounting for dynamic electron correlation is essential to correctly predict the relative energies of 1B and 2A states. Despite the fact that 2A/1A and 1B/2A conical intersections complicate the energy landscape by adding an extra dimension, the qualitative interpretation given by the state correlation diagram along symmetric coordinate still holds. It was further confirmed at the high level (CASPT2 and MRCI) by building *ab initio* PESs and performing two-dimensional quantum dynamics [36, 37] on these PESs. Barrierless descent on excited state PES was found to determine ultrafast photoconversion between CHD and HT, and quantum yield of this process was primarily determined by location of the 2A/1A CIX.

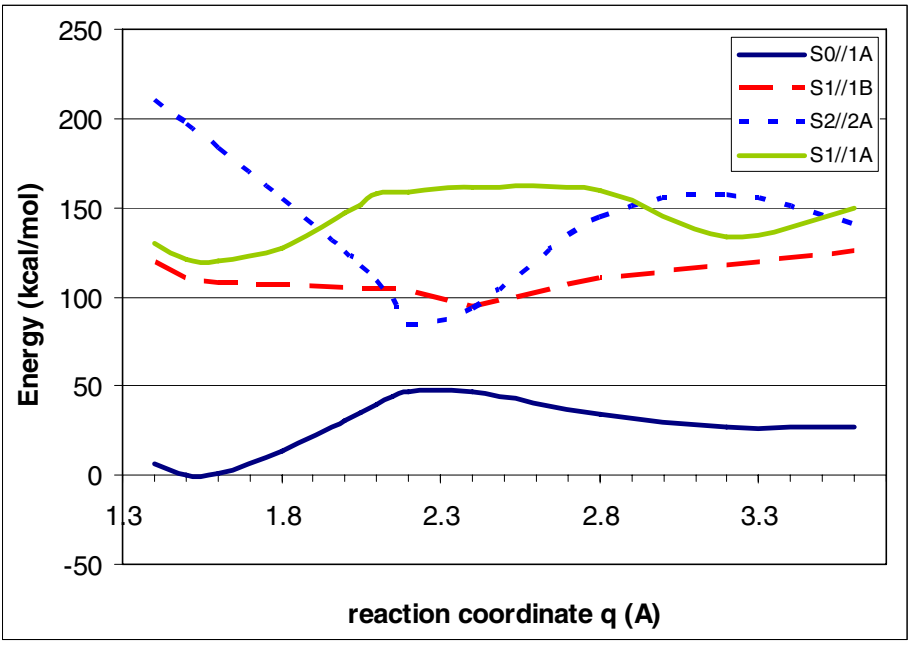


Fig. 1. Minimum energy pathways for the ground (S0, 1A), single excited (S1, 1B) and double excited states (S2, 2A) along the reaction pathway of the ring-closing C-C bond in the CHD/HT system, predicted at the ATDA-UDFT/6-31G//STS-DFT/6-31G theory level, using the M05-2X exchange-correlation functional. Absence of an appreciable energy barrier on the pathway from the 1B state of CHD in the Franck-Condon geometry (left) to the minimum on the 2A surface is consistent with ultrafast rate of the photoinitiated cycloreversion reaction CHD→HT.

Minimum energy pathways (MEPs) obtained in this study are plotted in Fig. 1. As the C1…C6 reaction coordinate contracts from a non-bonding distance to the normal covalent bond, the bright 1B state (characterized by the large transition dipole from the ground sate), is being monotonically stabilized in energy, starting descent to the pericyclic minimum. At the same time, the dark 2A state (with a negligibly small

transition dipole from the ground state) is being stabilized even faster, crosses the 1B state surface, and forms the bottom of the pericyclic minimum.

It is worth noting that geometry of the ground state does not approximate the excited state geometry accurately enough to produce a reasonable potential energy surface. The vertical excitation curve, plotted in Fig. 1 and representing energy of the 1B state at the ground state geometry, displays a maximum instead of a pericyclic minimum, and contradicts both high level *ab initio* and more accurate relaxed ATDA-UDFT data.

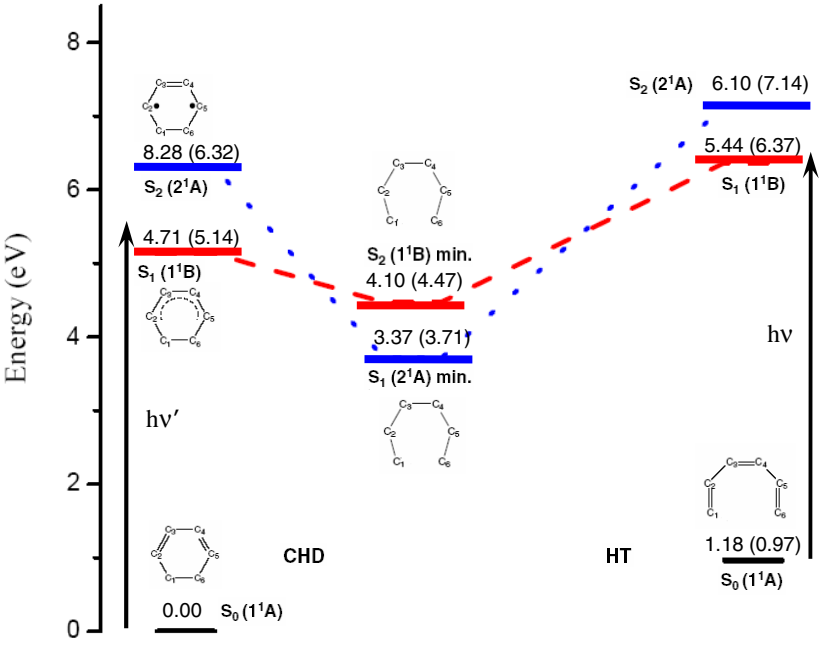


Fig. 2. Relative state energies (in eV) for the ground and two lowest singlet excited states in CHD/HT system, obtained at the UDFT/6-31G//STS-DFT/6-31G theory level, using the M05-2X exchange-correlation functional. The results of high level MR-PT2 *ab initio* calculations from Ref. [36] are shown for comparison in parentheses. Absence of an appreciable energy barrier on the pathway from the 1B state of CHD in the Franck-Condon geometry (left) to the minimum on the 2A surface explains ultrafast rate of the photoinitiated cycloreversion reaction CHD→HT.

A qualitative comparison between our modified DFT results and state-of-the-art wavefunction theory method MR-PT2 calculations of the CHD/HT system is presented in Fig. 2. Five important points were considered: ground state equilibrium geometries for closed and open isomers (CHD and HT), corresponding to Frank-Condon geometries of the ground states; and excited states 1B and 2A, optimized into the respective pericyclic minima. They are in surprisingly good agreement with high level *ab initio* results. As one can see, ATDA-UDFT //STS-DFT at the M05-2X/6-31G theory level, adopted in this work, almost uniformly overstabilizes

both excited states by 0.4-1.0 eV, but retains the correct order of excited states, as compared to the multireference perturbation theory results. To the best of our knowledge, this is the first report of the correct state ordering in this system, obtained from a DFT-based approach.

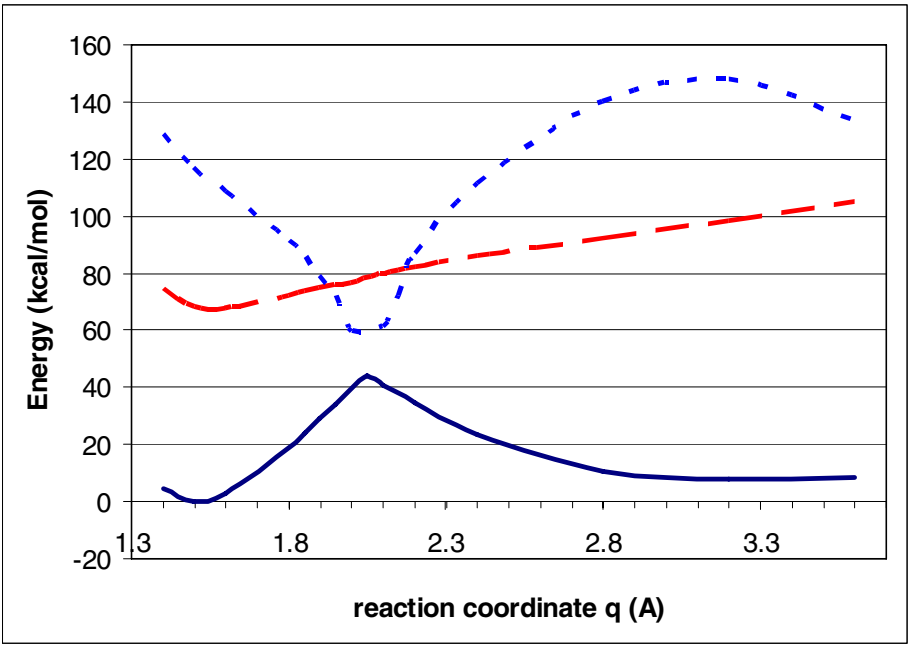


Fig. 3. Minimum energy pathways for the ground (S0, 1A), single excited (S1, 1B) and double excited states (S2, 2A) along the reaction pathway of ring-closing C–C bond in the dithioarylethene system, predicted at the ATDA-UDFT/6-31G//STS-DFT/6-31G theory level, using the M05-2X exchange-correlation functional. The legend is the same as in Fig. 1. Absence of an appreciable energy barrier on the pathway from the 1B state of the open form in the Franck-Condon geometry (left) to the minimum on the 2A surface is consistent with ultrafast rate of the photoinitiated cycloreversion reaction. The distance C1-C6 (reaction coordinate) was kept frozen during geometry optimizations of the ground and excited states using the STS-DFT method.

Dithienylethene (Scheme 1, with X=S) is the simplest homolog of diarilethens, an important class of compounds for photoswitching applications. Our calculations produced the minimum energy pathways for the ground and the two lowest single and double excited states, which are plotted in Fig. 3. One can see that MEP of the 1B state has a minimum in the closed form, and monotonically rises to the Franck-Condon region of the open form. The doubly excited 2A state, on the other hand, forms a pericyclic minimum and crosses below the 1B state in the vicinity of the conical intersection.

Therefore, the excitation of the open form is followed by the ultrafast barrierless relaxation into pericyclic minimum along 1B and then 2A PES, while the excitation

of the closed form will populate the potential minimum on the excited state surface. The conversion of the excited closed form into the pericyclic minimum must first overcome the small (c.a. 5 kcal/mol) potential energy barrier, which leads to relatively slow cycloreversion. The excited state absorption will then bring the system from the 1B to the 2A state, followed by barrierless relaxation toward CIX. Thus, our MEPs explain both slow cycloreversion and ultrafast photoswitching upon sequential two-photon absorption.

5 Conclusions

A new approach to plot potential energy surfaces of the excited states, based on Density Functional Theory is presented. This approach includes both single and double excitations appearing in first and second order Time-Dependent DFT in the Coupled Electronic Oscillator formalism (dubbed the *a posteriori* Tamm-Dancoff approximation, ATDA-DFT). Unphysical spikes on these surfaces close to pericyclic minima were traced to the failure of the restricted Kohn-Sham formalism to describe the partial bond breaking on the ground states, and were eliminated by replacing the ground state energy component of the excited state with the one obtained in the unrestricted broken symmetry Kohn-Sham formalism (termed here ATDA-UDFT). Importance of excited state geometry optimization (as opposed to the habitual use of unrelaxed ground state geometries) in accurate prediction of these potential energy surfaces was demonstrated. For the lack of analytical derivatives at the ATDA-UDFT theory level the lowest single and double excited state geometry is approximated using the Slater Transition State method (STS-DFT). The combined ATDA-UDFT//STS-DFT approach was shown to slightly underestimate energy of both excited states but correctly reproduce the state ordering and the energy crossovers as compared to the high-level multireference perturbation theory results for hexatriene/cyclohexadiene system. The approach was also able to explain experimentally observed slow photochemical cycloreversion rates and fast excited state absorption initiated cycloreversion in model dithioarylethenes. This method may assist in future development of new photoswitching materials for advanced applications in the Information Technology.

Acknowledgements. This work is supported in part by the National Science Foundation (CCF 0740344). The authors are thankful to UCF Institute for Simulations and Training HPC Stokes facility, UCF I2Lab, and DOE NERSC for the generous donation of computer time. AEM acknowledges the ACS COMP Hewlett-Packard Outstanding Junior Faculty award presented for this work at the Fall 2008 American Chemical Society Meeting.

References

1. Kaneko, M.: Materials for magneto-optical recording. Mrs Bulletin 31, 314–317 (2006)
2. Wang, J.G., Sun, C.J., Hashimoto, Y., Kono, J., Khodaparast, G.A., Cywinski, L., Sham, L.J., Sanders, G.D., Stanton, C.J., Munekata, H.: Ultrafast magneto-optics in ferromagnetic III-V semiconductors. Journal of Physics-Condensed Matter 18, R501–R530 (2006)

3. Zhou, G.F.: Materials aspects in phase change optical recording. Materials Science and Engineering a-Structural Materials Properties Microstructure and Processing 304, 73–80 (2001)
4. Kawata, S., Kawata, Y.: Three-dimensional optical data storage using photochromic materials. Chemical Reviews 100, 1777–1788 (2000)
5. Irie, M.: Diarylethenes for memories and switches. Chemical Reviews 100, 1685–1716 (2000)
6. Ern, J., Bens, A.T., Martin, H.D., Mukamel, S., Tretiak, S., Tsyganenko, K., Kuldova, K., Trommsdorff, H.P., Kryschi, C.: Reaction Dynamics of a Photochromic Fluorescing Dithienylethene. The Journal of Physical Chemistry A 105, 1741–1749 (2001)
7. Nakamura, S., Yokojima, S., Uchida, K., Tsujioka, T., Goldberg, A., Murakami, A., Shinoda, K., Mikami, M., Kobayashi, T., Kobatake, S., Matsuda, K., Irie, M.: Theoretical investigation on photochromic diarylethene: A short review. Journal of Photochemistry and Photobiology A: Chemistry 200, 10–18 (2008)
8. Garavelli, M.: Computational organic photochemistry: strategy, achievements and perspectives. Theoretical Chemistry Accounts 116, 87–105 (2006)
9. Truhlar, D.G., Gordon, M.S.: From Force-Fields to Dynamics - Classical and Quantal Paths. Science 249, 491–498 (1990)
10. Fuss, W., Hering, P., Kompa, K.L., Lochbrunner, S., Schikarski, T., Schmid, W.E., Trushin, S.A.: Ultrafast photochemical pericyclic reactions and isomerizations of small polyenes. Berichte Der Bunsen-Gesellschaft-Physical Chemistry Chemical Physics 101, 500–509 (1997)
11. Bernardi, F., Olivucci, M., Robb, M.A.: Potential energy surface crossings in organic photochemistry. Chemical Society Reviews 25, 321–328 (1996)
12. Fuss, W., Lochbrunner, S., Muller, A.M., Schikarski, T., Schmid, W.E., Trushin, S.A.: Pathway approach to ultrafast photochemistry: potential surfaces, conical intersections and isomerizations of small polyenes. Chemical Physics 232, 161–174 (1998)
13. Robb, M.A., Garavelli, M., Olivucci, M., Bernardi, F.: A computational strategy for organic photochemistry. Reviews in Computational Chemistry 15, 87–146 (2000)
14. Toniolo, A., Ben-Nun, M., Martinez, T.J.: Optimization of conical intersections with floating occupation semiempirical configuration interaction wave functions. Journal of Physical Chemistry A 106, 4679–4689 (2002)
15. Yarkony, D.R.: Marching along ridges. An extrapolatable approach to locating conical intersections. Faraday Discussions 127, 325–336 (2004)
16. Runge, E., Gross, E.K.U.: Density-Functional Theory for Time-Dependent Systems. Physical Review Letters 52, 997–1000 (1984)
17. Levine, B.G., Ko, C., Quenneville, J., Martinez, T.J.: Conical intersections and double excitations in time-dependent density functional theory. Molecular Physics 104, 1039–1051 (2006)
18. Gunnarsson, O., Lundqvist, B.I.: Exchange and Correlation in Atoms, Molecules, and Solids by Spin-Density Functional Formalism. Physical Review B 13, 4274–4298 (1976)
19. Cai, Z.L., Reimers, J.R.: Application of time-dependent density-functional theory to the (3)Sigma(-)(u) first excited state of H-2. Journal of Chemical Physics 112, 527–530 (2000)
20. Casida, M.E., Ipatov, A.: Excited-state spin-contamination in time-dependent density-functional theory for molecules with open-shell ground states. Abstr. Pap. Am. Chem. Soc. 231, 94-COMP (2006)
21. Neugebauer, J., Baerends, E.J., Nooijen, M.: Vibronic coupling and double excitations in linear response time-dependent density functional calculations: Dipole-allowed states of N-2. Journal of Chemical Physics 121, 6155–6166 (2004)

22. Knoester, J., Mukamel, S.: Nonlinear optics using the multipolar hamiltonian - the bloch-maxwell equations and local-fields. Physical Review A 39, 1899–1914 (1989)
23. Tretiak, S., Mukamel, S.: Density matrix analysis and simulation of electronic excitations in conjugated and aggregated molecules. Chemical Reviews 102, 3171–3212 (2002)
24. Tretiak, S., Chernyak, V.: Resonant nonlinear polarizabilities in the time-dependent density functional theory. Journal of Chemical Physics 119, 8809–8823 (2003)
25. Mikhailov, I.A., Tafur, S., Masunov, A.E.: Double excitations and state-to-state transition dipoles in pi-pi* excited singlet states of linear polyenes: Time-dependent density-functional theory versus multiconfigurational methods. Physical Review A 77, 012510–012511 (2008)
26. Slater, J.C.: Advances in Quantum Chemistry 6, 1–92 (1972)
27. Liberman, D.A.: Slater transition-state band-structure calculations. Physical Review B 62, 6851–6853 (2000)
28. Noodleman, L., Baerends, E.J.: Electronic-structure, magnetic-properties, electron-spin-resonance, and optical-spectra for 2-fe ferredoxin models by lcao-x-alpha valence bond theory. Journal of the American Chemical Society 106, 2316–2327 (1984)
29. Hu, C., Sugino, O.: Average excitation energies from time-dependent density functional response theory. The Journal of Chemical Physics 126, 074112–074110 (2007)
30. Han, W.G., Liu, T.Q., Lovell, T., Noodleman, L.: Density functional theory study of Fe(IV) d-d optical transitions in active-site models of class I ribonucleotide reductase intermediate X with vertical self-consistent reaction field method. Inorganic Chemistry 45, 8533–8542 (2006)
31. Frisch, M.J., et al.: Gaussian 2003. Gaussian, Inc., Wallingford (2004)
32. Kuthirummal, N., Rudakov, F.M., Evans, C.L., Weber, P.M.: Spectroscopy and femtosecond dynamics of the ring opening reaction of 1,3-cyclohexadiene. The Journal of Chemical Physics 125, 133307–133308 (2006)
33. Celani, P., Ottani, S., Olivucci, M., Bernardi, F., Robb, M.A.: What Happens During the Picosecond Lifetime of 2a(1) Cyclohexa-1,3-Diene - a Cas-Scf Study of the Cyclohexadiene Hexatriene Photochemical Interconversion. Journal of the American Chemical Society 116, 10141–10151 (1994)
34. Celani, P., Bernardi, F., Robb, M.A., Olivucci, M.: Do photochemical ring-openings occur in the spectroscopic state? B-1(2) pathways for the cyclohexadiene/hexatriene photochemical interconversion. Journal of Physical Chemistry 100, 19364–19366 (1996)
35. Garavelli, M., Celani, P., Fato, M., Bearpark, M.J., Smith, B.R., Olivucci, M., Robb, M.A.: Relaxation paths from a conical intersection: The mechanism of product formation in the cyclohexadiene/hexatriene photochemical interconversion. Journal of Physical Chemistry A 101, 2023–2032 (1997)
36. Tamura, H., Nanbu, S., Nakamura, H., Ishida, T.: A theoretical study of cyclohexadiene/hexatriene photochemical interconversion: multireference configuration interaction potential energy surfaces and transition probabilities for the radiationless decays. Chemical Physics Letters 401, 487–491 (2005)
37. Tamura, H., Nanbu, S., Ishida, T., Nakamura, H.: Ab initio nonadiabatic quantum dynamics of cyclohexadiene/hexatriene ultrafast photoisomerization. The Journal of Chemical Physics 124, 084313 (2006)

Predictions of Two Photon Absorption Profiles Using Time-Dependent Density Functional Theory Combined with SOS and CEO Formalisms

Sergio Tafur[1,2], Ivan A. Mikhailov[1], Kevin D. Belfield[3,4], and Artëm E. Masunov[1,2,3]

[1] Nanoscience Technology Center
[2] Department of Physics
[3] Department of Chemistry
[4] CREOL, College of Optics and Photonics, University of Central Florida,
12424 Research Parkway, Suite 400, Orlando 32816, USA
amasunov@mail.ucf.edu

Abstract. Two-photon absorption (2PA) and subsequent processes may be localized in space with a tightly focused laser beam. This property is used in a wide range of applications, including three dimensional data storage. We report theoretical studies of 5 conjugated chromophores experimentally shown to have large 2PA cross-sections. We use the Time Dependent Density Functional Theory (TD-DFT) to describe the electronic structure. The third order coupled electronic oscillator formalism is applied to calculate frequency-dependent second order hyperpolarizability. Alternatively, the sum over states formalism using state-to-state transition dipoles provided by the *a posteriori* Tamm-Dancoff approximation is employed. It provides new venues for qualitative interpretation and rational design of 2PA chromophores.

Keywords: conjugated chromophores, two-photon absorption, time-dependent density functional theory, coupled electronic oscillators, sum over states, Tamm-Dancoff approximation, structure-activity relationship.

1 Introduction

Two-photon absorption (2PA) is an electronic excitation process involving simultaneous absorption of two photons. There are a wide range of 2PA applications, such as three dimensional data storage, photonic devices, lithographic micro-fabrication [1], quantum information technology [2], optical limiting, two-photon pumped lasing in organic chromophores and quantum dots [2, 3], in-vivo bioimaging, and cell-selective photo-dynamic therapy [3]. Most applications require chromophores with large 2PA cross-sections to minimize laser intensity requirements and prevent overheating of targets [1]. To design more efficient 2PA chromophores, it is important to understand their structure/activity relationships (SARs). Computer modeling of 2PA spectra facilitates understanding of these relationships and is becoming an important part rational approach that may accelerate progress in chromophore design [4]. Accurate predictions of 2PA spectral profiles would greatly assist in the design of more effective 2PA chromophores while eliminating poor candidates form the synthetic pipeline. The goal

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 179–188, 2009.

of this study is improvement in quantitative predictions of 2PA, as well as development of qualitative tools to understand the relations between the electronic structure of the chromophores and 2PA profiles.

In the past decades several research groups had made a strong effort aimed at the development of new compounds with large 2PA cross sections. The main guiding principle used in those studies involved electron transfer between electron-donor (D) and electron-acceptor (A) moieties attached symmetrically or asymmetrically to the π-conjugated bridge. Fluorene fragment in particular was found to be a good example of π-conjugated bridge due to highly delocalized π-system delocalized over the two benzene rings held together at nearly coplanar orientation by methylene bridge [5]. *D*-π-*A*, *D*-π-*D*, or *A*-π-*A* molecular structures have been proposed and studied both theoretically and experimentally. In recent studies fluorene derivatives have been extended to *D*-π-π-*A* and *A*-π-π-*A* types with the aim of increasing 2PA absorption cross-sections [6-10]. However, the choice of functional groups and linkages the most appropriate for developing chromophores with the largest 2PA characteristics it is still under active investigation.

In order to accelerate the experimental efforts based on traditional trial and error approach, a quantitative understanding of the trends in dependence of 2PA cross-section on molecular structure would be clearly beneficial. Two major approaches had been used applied to accomplish that goal. First is based on essential state models (three-state, four-state, etc.). Parameters of these models (such as excitation energies and transition dipoles) are adjusted to fit experimental data. These parameters are then correlated with details of molecular structure (π-conjugated chain lengths, donor/acceptor strengths, etc.). Another approach consists of quantitative prediction of 2PA cross-sections at chosen level of theory, followed by analysis of the physical principles of the major contributions into this property. The levels of theory, used for 2PA predictions cover the wide range.

In recent quantum chemical calculations performed on conjugated chromophores have shown that a substantial symmetric charge redistribution that occurs upon excitation may account for heightened sensitivity to 2PA events [6]. In their work Bredas *et al*. established a good agreement between the peak values of 2PA crossections measured with femto-second pulses and those calculated with semi-empirical intermediate neglect of differential overlap Hamiltonian with multi-reference double-configuration interaction (INDO-MRD-CI) scheme based methods. Aside from the donor-acceptor configuration of 2PA active chromophores, it was also established by Bredas *et al.* that increasing the length and charge transfer of the molecules results in an increase in 2PA crossections and may also result in a significant shift of 2PA to longer wavelengths [6]. Complementarily, Agren *et al*. theoretically studied four lowest excited states of π conjugated systems experimentally produced and characterized by Kim *et al.* [10] and Ventelon *et al.* [9] using *ab initio* response theory. They showed that their theoretical findings were consistent with the correlation between large 2PA crossections and a π center, but that though the one photon absorption (1PA) spectra was strongly correlated to the molecular length this was not always the case for 2PA in the visible domain [5]. At around this time it was also established by Fabian *et al.* that spectral absorption features are reasonably well reproduced by the approximate semi-empirical MO-CI methods based on the NDO (ZINDO/S), however time-dependent density functional response theory (TD-DFT) proved to be superior over semi-empirical methods [11].

Since then Hales *et al.* showed that 2PA spectra for symmetric and asymmetric fluorene derivative compounds exhibit intermediate resonant enhancement of nonlinearities, with an order of magnitude enhancement for asymmetric cases, when compared to degenerate 2PA. INDO-MRD-CI semi-empirical methods that implemented a simplified three level model were also shown to provide additional insight into the mechanisms governing 2PA events [12]. Several groups published works investigating the structure-activity relationships (SARs) responsible for the 2PA characteristics.

The conjugated chromophores selected as the subjects of this study are presented in Scheme 1. Theoretical models of these were derived by truncation of the aliphatic chains and replacing them with to methyl groups in the original experimental structures. The abbreviations of the model molecules and the systematic names of the corresponding experimentally studied ones are: **BzFBz**: 2,7-Bisbenzothiazolyl-9,9-didecylfluorene; **BzFDp**: (7-benzothiazol-2-yl-9,9-didecylfluoren-2-yl)diphenylamine; **DpFDp**: 9,9-didecyl-2,7-bis(N,N-phenylamino)-fluorene; **BzPFPBz**: 2,7-Bis[4-(9,9-didecylfluoren-2-yl)vinyl]-phenylbenzothiazole; **DpPFPBz**: {7-[2-(4-Benzothiazol-2ylphenyl)vinyl]-9,9 didecylfluorene-2yl}diphenylamine. These compounds were experimentally synthesized and characterized by Belfield *et al.* as a model compounds for possible applications in two photon microfabrication, two photon photochemical transformations, non-destructive 3-D multiphoton fluorescence imaging, and photodynamic therapy [7, 12, 13]. They found large (600GM) cross-sections for **BzFBz** while studying the design of rigid-rod polymers while 2PA cross-sections of the polymers were reduced by aggregation [14]. Compound **BzFBz** additionally exhibited a large fluorescence quantum yield. The good chemical, thermal, and photochemical stability, combined with desirable one- and two-photon absorption and luminescence properties, stand out as characteristics of this chromophore as a promising material for two-photon based technologies [15]. Compound **BzFDp** has been previously implemented for in vivo 2PA biomedical imaging applications, as a fluorophore dye used for staining rat cardiomyoblast cells (H9c2), by Belfield *et al.* due to its high photostability, fluorescence quantum yield, and two-photon absorption cross-section over the tunable range of commercially available Ti:sapphire lasers [16]. Additionally, **BzFDp** has been investigated as a potential 2PA free-radical photo-initiator for three-dimensionally resolved polymerization, resulting in intricate micro-fabrication and imagining [17].

In semi-empirical wave function theory studies of 2PA active organics have been carried to a varying degree of success. The efforts put forward in these studies have circled around INDO (intermediate neglect of differential overlap) semi-empirical Hamiltonian models for molecular geometry optimizations and the implementation of the INDO Hamiltonian coupled to a MRD-CI (multi-reference double configuration interaction) formalism in the description of ground and excited states. The description of these states were then used to calculate ground and excited state energies, dipoles and transition dipoles [18] which in turn were implemented in the sum over states formalism (SOS) to calculate linear or nonlinear material response.

Recently, Time Dependent Density Functional Theory (TD-DFT) was successfully used to simulate 2PA electronic spectra in large conjugated molecules [19-21]. The coupled electronic oscillator (CEO) formalism performed well, after it was shown that the density matrix formulation of the time-dependent Kohn-Sham equations allows treatment of adiabatic TD-DFT on the same footing as the TDHF theory to an arbitrary order in the external perturbation [22]. This approach was shown to achieve superior

accuracy for 2PA excitation energies, when compared to semi-empirical wave function theory methods. The tools used for implementation of the third order CEO at TD-DFT level are detailed below.

An alternative approach to using CEO in the framework of TD-DFT is to use state to state transition dipole moments calculated within TD-DFT, using second and third order response functions, and implement them in the SOS formalism [23, 24]. Difficulties that arise in the implementation of SOS are governed by the description/accuracy of the state to state transition dipole moments and excited state energies predicted by DFT. Cronstrand *et al.* and Kamada *et al.* present the possibility of using few states models derived from SOS to calculate the 2PA cross-section and arrive at a mechanism for 2PA [25, 26].

Of the compounds in Table 1, **BzFDp** and **DpFDp** have been previously studied theoretically by Hales *et al.* using INDO-MRD-CI with geometries optimized with an AM1 semi-empirical Hamiltonian. All their calculations were carried out on isolated molecules and showed a strong qualitative and quantitative agreement with experimentally generated spectra [12]. The three-level model developed in their study provided further insight into the mechanisms governing the nonlinear activity of 2PA active chromophores in relation to the description of the molecular states. A second study carried out by Day *et al.* implemented linear and quadratic response density functional theories to calculate the photo-physical properties of D-π-A molecules including **BzFDp**. Their comparison of a two-state approximation and with calculation of 2PA via the SOS method with the inclusion of higher energy states drew a conclusion that the inclusion of higher energy states was necessary in the description of 2PA [27, 28].

In this contribution we obtain approximate state-to-state transition dipole moments μ_{nm}^{i} within a TD-DFT formalism by implementing the *a posteriori* Tamm-Dancoff approximation (ATDA, introduced in Ref. [29]), as an approximation to second order DFT, and employ them to identify the essential states governing the 2PA process. We also validate ATDA results by using these μ_{nm}^{i} to evaluate the resonant 2PA cross-sections with sum over state models (SOS) and compare them to CEO results as well as experimental values.

2 Theory

Time-Dependent Density Functional Theory (TD-DFT) was recently combined with the Coupled Electronic Oscillator (CEO) formalism to simulate 2PA electronic spectra in large conjugated molecules [19, 30]. These and other 2PA predictions using TD-DFT [31] were shown to achieve quantitative agreement with experiment and higher-level *ab initio* predictions. In this contribution we also use an alternative Sum Over States (SOS) approach, calculate state-to-state transition dipole moments μ_{nm}^{i} using the *a posteriori* Tamm-Dancoff approximation and employ them to identify essential states governing the 2PA process. We also validate ATDA by using these approximate μ_{nm}^{i} to predict resonant 2PA cross-sections with the SOS model and compare them to CEO results as well as experimental values.

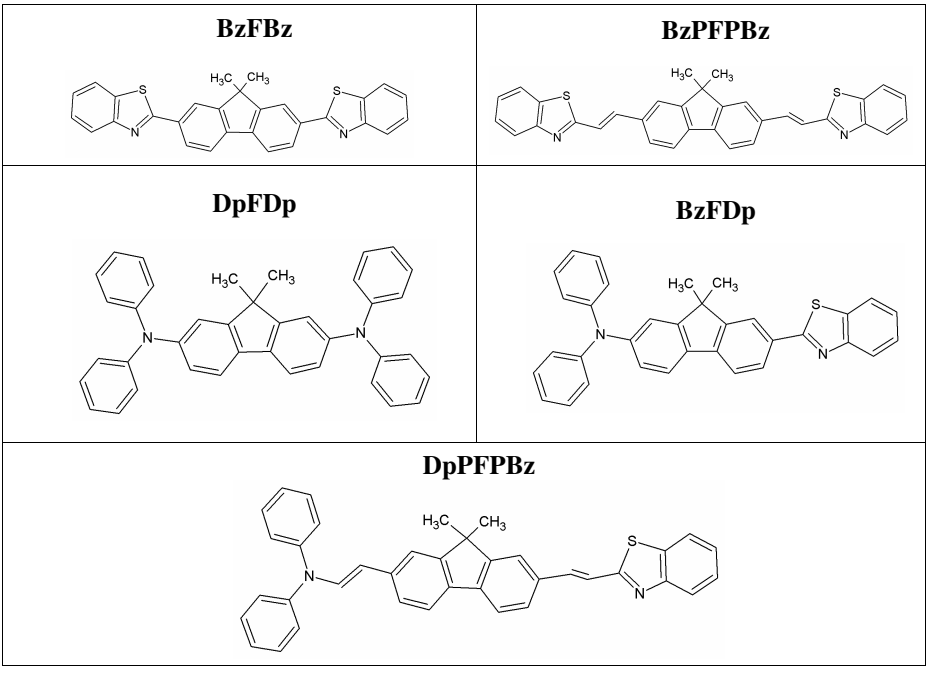
In the 2PA-transition matrix approximation the 2PA cross-section is given by [25]:

$$\sigma_{2PA} = \frac{8\pi^3\omega^2}{15c^2} g(2\omega) \sum_{a,b}^{3} \left(M_{aa} M_{bb}^{*} + 2M_{ab} M_{ab}^{*} \right),$$
$$M_{ij} = \frac{1}{2\hbar} \sum_{k}^{N} \left(\frac{\mu_{fk}^{i} \mu_{kg}^{j} + \mu_{fk}^{j} \mu_{kg}^{i}}{\omega_{kg} - \omega - i\Gamma/\hbar} \right) \quad (1)$$

here g(2ω) is the Lorentzian lineshape, and μ_{nm}^{i} are state-to-state transition dipole moments. This ATDA/SOS approach opens new venues for interpretation of 2PA properties in terms of molecular electronic structure and can be used for rational design of 2PA chromophores.

3 Computational Details

The chromophore molecules selected for this study are presented in Scheme 1. They were derived from experimentally studied ones by truncation of the aliphatic chains to methyl group. All molecular structures were optimized at HF/STO-3G theory level, which favors planar geometry of conjugated molecules and was shown [31] to give the best agreement for the bond lengths as compared to the results of X-Ray diffraction experiments for stylbene and its three derivatives. The optimized geometries were confirmed by the absence of imaginary frequencies in the following normal mode calculations. The single point energy and transition dipole calculations were performed at the TD-B3LYP/MIDIx level.



Scheme 1. Structural formulas of the molecules studied

Transition density matrices for the lowest 24 excited states, as well as Kohn-Sham operators on these transition densities were printed out. Contributions of the second and third derivatives of the exchange-correlation potentials into Kohn-Sham operators, and operators on the pair combinations of transition densities were neglected. Commercially available computational program Gaussian98 [32] was modified as described in previous studies [19] in order to enable this printout. The frequency-dependent orientationally averaged first- and third-order polarizability tensors were generated from the generated matrices using (1) and expressions implemented in CEO program [33]. The habitual empirical linewidth of 0.1 eV was used for both 1PA and 2PA. To analyze the electronic structure of the excited states we used natural transition orbitals (NTO), which diagonalize the transition density matrix, and give the best representation of the electronic excitation in single-particle terms [34]. Graphical software XCrysDen [35] was used to plot NTOs.

4 Results and Discussion

We present 2PA resonant energies and cross-sections in Table 1. For two of the molecules, the profiles obtained with both SOS and CEO formalisms are presented in Fig. 1, along with linear spectra, and natural transition orbitals.

Table 1. Energies and cross-sections for the linear and 2PA absorbing states in the molecules studied

State	$2PA_{calc}$,GM	$\Delta E_{vertical}$, eV	$2PA_{exp}$,GM	ΔE_{exp}, eV	λ_{exp}, nm
BzFBz					
S1	-	3.59	-	3.41	364
S4	324	4.28	437	4.27	290
BzFDp					
S1	65	3.08	73	3.21	387
S3	151	4.10	-	-	-
S4	151	4.12	-	-	-
DpFDp					
S1	-	3.38	-	3.10	400
S3	126	3.88	89	4.00	310
S5,6	-	4.01	-	-	-
S15	162	4.60	-	-	-
BzPFPBz					
S1	-	3.06	-	3.08	403
S2	711	3.46	-	-	-
S4	-	3.99	-	4.00	310
S11	486	4.45	-	-	-
DpPFPBz					
S1	154	2.83	162	3.24	383
S2	133	3.56	-	4.03	306
S6,7,8	445	4.25	-	-	-

The SOS and CEO results (marked by solid lines on 2PA spectra in Fig.1) are in excellent quantitative agreement with each other, which provides a validation for the ATDA/SOS method. Predicted 2PA profiles also agree well with experimental ones. Experimental measurements of their spectral properties were reported in [14-16, 36]. While for most molecules agreement between resonant maxima is better than 0.1 eV, in the case of **DpPFPBz** theoretical bands are red-shifted relative to experimental

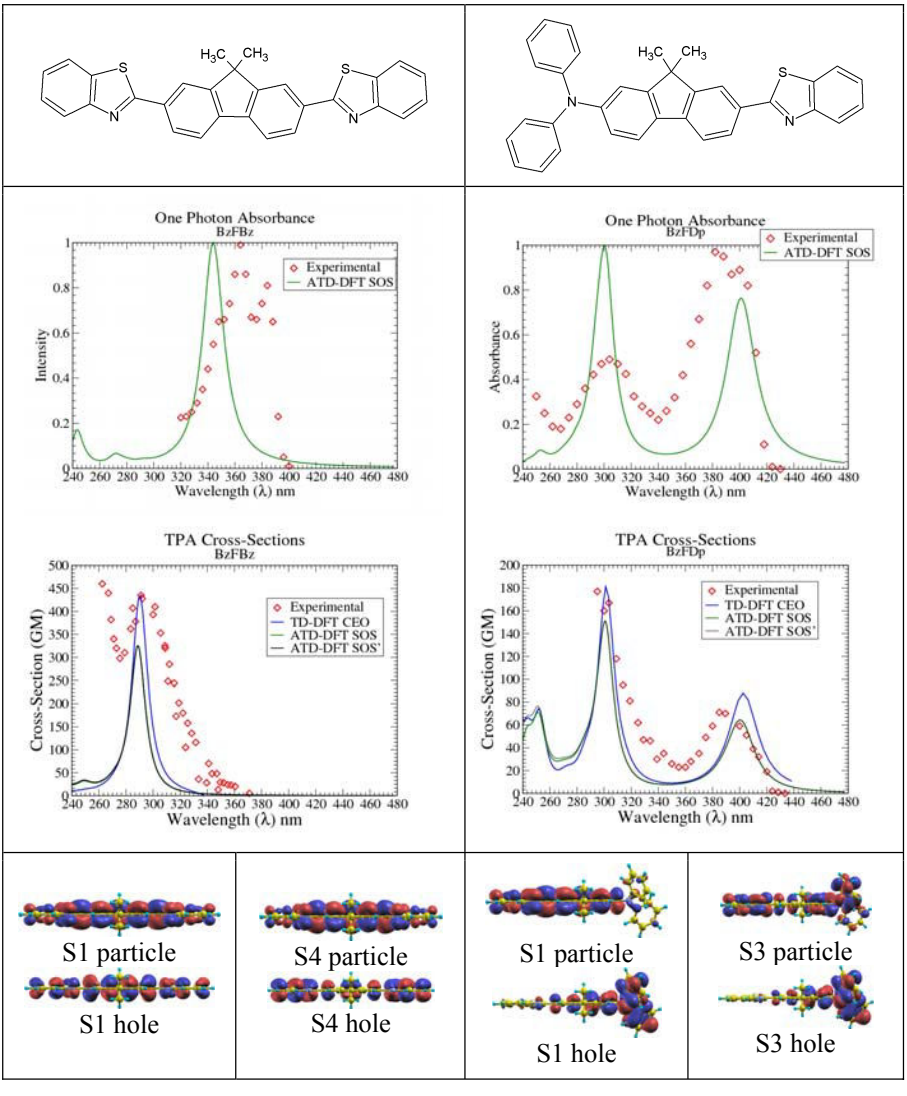


Fig. 1. Structural formulas *(top row)*, 1PA profiles *(row 2)*, 2PA profiles *(row 3)*, and isosurfaces for natural transition orbitals *(bottom row)* for studied conjugated chromophores. Diamonds mark the experimental profiles; solid lines correspond to theoretical predictions with SOS and CEO formalisms.

ones by approximately 0.44 eV (55 nm). This can be explained by greater conformational flexibility of the longer conjugated chain, and blue shift of the absorption spectra for non-planar conformations. Overall, agreement with experiment validates the use of TD-DFT as a part of rational design strategies directed toward new and improved two-photon absorbing materials.

5 Conclusions

We report theoretical study of five conjugated chromophores experimentally shown to have large two photon absorption cross-sections. We use the third order response formalism within Time Dependent Density Functional Theory to calculate frequency-dependent second order hyperpolarizability in both the sum over states and coupled electronic oscillator formalisms to describe 2PA cross-sections. While CEO expressions do not lend themselves easily to a qualitative analysis, SOS ones can be simplified to essential state models and employed to identify 2PA resonant states and interpret the relationships between electronic structure and 2PA profiles.

We also use Natural Transition orbitals to compare the electronic structure of the linear and two-photon absorbing states. State to state transition dipole moments, necessary for SOS expressions are calculated with the *a posteriori* Tamm-Dancoff approximation and used to describe two-photon processes. Numerical values of the cross-sections obtained in SOS and CEO were found to be in good quantitative agreement with each other. This is the first time that TD-DFT/CEO and ATDA-DFT/SOS methods have been compared for the calculation of 2PA spectra. Both CEO and SOS results are in good agreement with experiment. This validates the use of TD-DFT as a part of rational design strategies directed toward new and improved Two-Photon absorbing materials for bioimaging and optical data storage.

Acknowledgments. This work was supported in part by the National Science Foundation Grant No. CCF 0740344. The authors are thankful to DOE NERSC, UCF I2Lab, and UCF Institute for Simulations and Training (IST) HPC Stokes facility for the generous donation of the computer time. ST acknowledges support by the UCF NSTC and UCF Graduate Studies Office through the Summer Mentoring Research Fellowship. AEM acknowledges ACS COMP Hewlett-Packard Outstanding Junior Faculty award presented for this work at the Fall 2008 American Chemical Society Meeting.

References

1. Cumpston, B.H., et al.: Two-photon polymerization initiators for three-dimensional optical data storage and microfabrication. Nature 398, 51–54 (1999)
2. Kagotani, Y., et al.: Two-photon absorption and lasing due to biexciton in CuCl quantum dots. Journal of Luminescence 112, 113–116 (2005)
3. Zipfel, W.R., et al.: Live tissue intrinsic emission microscopy using multiphoton-excited native fluorescence and second harmonic generation. Proceedings of the National Academy of Sciences of the United States of America 100, 7075–7080 (2003)

4. Rumi, M., et al.: Structure-property relationships for two-photon absorbing chromophores: Bis-donor diphenylpolyene and bis(styryl)benzene derivatives. Journal of the American Chemical Society 122, 9500–9510 (2000)
5. Wang, C.K., et al.: Effects of pi centers and symmetry on two-photon absorption cross sections of organic chromophores. Journal of Chemical Physics 114, 9813–9820 (2001)
6. Albota, M., et al.: Design of organic molecules with large two-photon absorption cross sections. Science 281, 1653–1656 (1998)
7. Belfield, K.D., et al.: Multiphoton-absorbing organic materials for microfabrication, emerging optical applications and non-destructive three-dimensional imaging. Journal of Physical Organic Chemistry 13, 837–849 (2000)
8. Belfield, K.D., et al.: Synthesis, characterization, and optical properties of new two-photon-absorbing fluorene derivatives. Chemistry of Materials 16, 4634–4641 (2004)
9. Ventelon, L., Moreaux, L., Mertz, J., Blanchard-Desce, M.: New quadrupolar fluorophores with high two-photon excited fluorescence. Chemical Communications, 2055–2056 (1999)
10. Kim, O.K., et al.: New class of two-photon-absorbing chromophores based on dithienothiophene. Chemistry of Materials 12, 284–286 (2000)
11. Fabian, J., et al.: Calculation of excitation energies of organic chromophores: a critical evaluation. Journal of Molecular Structure-Theochem 594, 41–53 (2002)
12. Hales, J.M., Hagan, D.J., Van Stryland, E.W., Schafer, K.J., Morales, A.R., Belfield, K.D., Pacher, P., Kwon, O., Zojer, E., Bredas, J.L.: Resonant enhancement of two-photon absorption in substituted fluorene molecules. Journal of Chemical Physics 121, 3152–3160 (2004)
13. Belfield, K.D., et al.: One- and two-photon fluorescence anisotropy of selected fluorene derivatives. Journal of Fluorescence 15, 3–11 (2005)
14. Belfield, K.D., Yao, S., Morales, A.R., Hales, J.M., Hagan, D.J., Van Stryland, E.W., Chapela, V.M., Percino, J.: Synthesis and characterization absorbing polymers of novel rigid two-photon. Polymers for Advanced Technologies 16, 150–155 (2005)
15. Belfield, K.D., et al.: Excited-state absorption and anisotropy properties of two-photon absorbing fluorene derivatives. Applied Optics 44, 7232–7238 (2005)
16. Schafer-Hales, K.J., Belfield, K.D., Yao, S., Frederiksen, P.K., Hales, J.M., Kolattukudy, P.E.: Fluorene-based fluorescent probes with high two-photon action cross-sections for biological multiphoton imaging applications. Journal of Biomedical Optics 10, 8 (2005)
17. Belfield, K.D., Schafer, K.J.: A new photosensitive polymeric material for WORM optical data storage using multichannel two-photon fluorescence readout. Chemistry of Materials 14, 3656–3662 (2002)
18. Kogej, T., Beljonne, D., Meyers, F., Perry, J.W., Marder, S.R., Bredas, J.L.: Mechanisms for enhancement of two-photon absorption in donor-acceptor conjugated chromophores. Chemical Physics Letters 298, 1–6 (1998)
19. Masunov, A.M., Tretiak, S.: Prediction of two-photon absorption properties for organic chromophores using time-dependent density-functional theory. Journal of Physical Chemistry B 108, 899–907 (2004)
20. Chen, G.H., Mukamel, S., Beljonne, D., Bredas, J.L.: The coupled electronic oscillators vs the sum-over-states pictures for the optical response of octatetraene. Journal of Chemical Physics 104, 5406–5414 (1996)
21. Cronstrand, P., Norman, P., Luo, Y., Agren, H.: Few-states models for three-photon absorption. Journal of Chemical Physics 121, 2020–2029 (2004)
22. Tretiak, S., Chemyak, V.: Resonant nonlinear polarizabilities in the time-dependent density functional theory. Journal of Chemical Physics 119, 8809–8823 (2003)

23. Gel'mukhanov, F., Baev, A., Macak, P., Luo, Y., Agren, H.: Dynamics of two-photon absorption by molecules and solutions. Journal of the Optical Society of America B-Optical Physics 19, 937–945 (2002)
24. Baev, A., Prasad, P.N., Samoc, M.: Ab initio studies of two-photon absorption of some stilbenoid chromophores. The Journal of Chemical Physics 122, 224309–224306 (2005)
25. Ohta, K., Kamada, K.: Theoretical investigation of two-photon absorption allowed excited states in symmetrically substituted diacetylenes by ab initio molecular-orbital method. Journal of Chemical Physics 124, 124303–124311 (2006)
26. Cronstrand, P., Luo, Y., Agren, H.: Generalized few-state models for two-photon absorption of conjugated molecules. Chemical Physics Letters 363, 198 (2002)
27. Day, P.N., Nguyen, K.A., Pachter, R.: TDDFT study of one- and two-photon absorption properties: Donor-pi-acceptor chromophores. Journal of Physical Chemistry B 109, 1803–1814 (2005)
28. Day, P.N., Nguyen, K.A., Pachter, R.: Calculation of two-photon absorption spectra of donor-pi-acceptor compounds in solution using quadratic response time-dependent density functional theory. Journal of Chemical Physics 125, 094103–094113 (2006)
29. Mikhailov, I.A., Tafur, S., Masunov, A.E.: Double excitations and state-to-state transition dipoles in pi-pi* excited singlet states of linear polyenes: Time-dependent density-functional theory versus multiconfigurational methods. Physical Review A 77, 012510–012511 (2008)
30. Badaeva, E.A., Timofeeva, T.V., Masunov, A.M., Tretiak, S.: Role of donor-acceptor strengths and separation on the two-photon absorption response of cytotoxic dyes: A TD-DFT study. J. Phys. Chem. A 109, 7276–7284 (2005)
31. Cronstrand, P., et al.: Density functional response theory calculations of three-photon absorption. Journal of Chemical Physics 121, 9239–9246 (2004)
32. Frisch, M.J., et al.: Gaussian 98 Revision A.11 (1998)
33. Tretiak, S., Mukamel, S.: Density matrix analysis and simulation of electronic excitations in conjugated and aggregated molecules. Chemical Reviews 102, 3171–3212 (2002)
34. Martin, R.L.: Natural transition orbitals. Journal of Chemical Physics 118, 4775–4777 (2003)
35. Kokalj, A.: Computer graphics and graphical user interfaces as tools in simulations of matter at the atomic scale. Computational Materials Science 28, 155–168 (2003), http://www.xcrysden.org/
36. Belfield, K.D., Bondar, M.V., Hernandezt, F.E., Przhonska, O.V., Yao, S.: Two-photon absorption cross section determination for fluorene derivatives: Analysis of the methodology and elucidation of the origin of the absorption processes. Journal of Physical Chemistry B 111, 12723–12729 (2007)

The Kinetics of Charge Recombination in DNA Hairpins Controlled by Counterions

Gail S. Blaustein, Frederick D. Lewis, Alexander L. Burin, and Rajesh Shrestha

Departments of Chemistry, Tulane University, New Orleans, LA 70118 and Northwestern University, Evanston, Il 60208

Abstract. The charge recombination rate in DNA hairpins is investigated. The distance dependence for the charge recombination rate between stilbene donor (Sd^+) and stilbene acceptor (Sa^-) linkers separated by an AT bridge has a double exponential form. We suggest that this dependence is associated with two tunneling channels distinguished by the presence or absence of the Cl^- counterion bound to Sd^+. Experiment-based estimates of counterion binding parameters agree within reasonable expectations. A control experiment replacing the Cl^- ion with other halide ions is suggested. Counterion substitution allows modification of the charge recombination rate in either direction by orders of magnitude.

Keywords: DNA hairpin, charge transfer, counterions.

1 Introduction

Electronic excitation of DNA using various optical methods is important for investigating DNA structure and biological function [1]. It also plays a fundamental role in a variety of DNA applications in nanotechnology which have been extensively considered during the past decade [2], [3], [4], [5]. The first time-resolved observation of charge transfer in DNA was made using stilbene capped DNA hairpins (Fig.1) [6], which are used successfully in the investigation of DNA electronic excitations and their kinetics. It is remarkable that hairpins with poly-A poly-T bridges connecting stilbene acceptor (Sa) and donor (Sd) linkers (Fig.1) can possess extremely long recombination times for the charge separated state $Sa^-(AT)_nSd^+$ following Sa photoexcitation. Increasing the number of AT pairs forming the bridge from $n = 1$ to $n = 7$ reduces the charge recombination rate by eight orders of magnitude [7]. This interesting property of stilbene capped DNA hairpins has potential for a variety of applications involving charge separation such as solar cells [8].

Therefore, it is important to understand mechanisms of charge recombination in DNA hairpins and investigate possible ways to control this process. This requires understanding distance dependence of the charge transfer rate (1).

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 189–196, 2009.

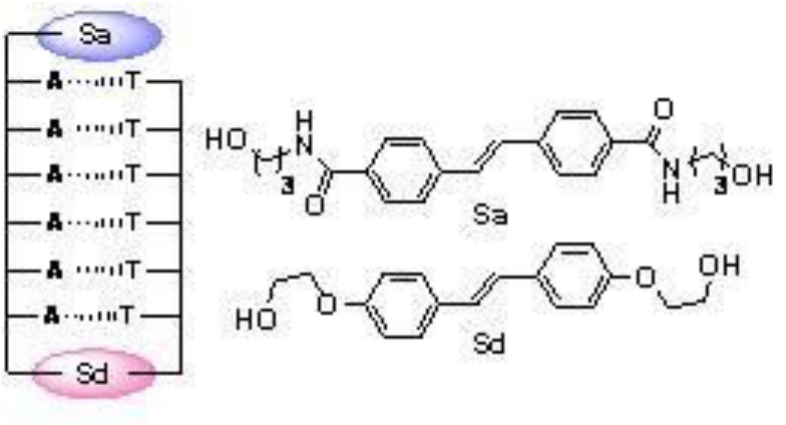


Fig. 1. A hairpin with six AT base pairs and the stilbene linkers

2 Kinetics Model

In this paper, the charge recombination rate in poly-A poly-T DNA hairpins [6] of lengths $n = 1$ to $n = 7$ is investigated. It is shown that the charge recombination rate for AT bridges as a function of distance r between Sd^+ and Sa^- has the double exponential form

$$k\,(r) = k_0\left[\exp\left(-\beta_1 r\right) + C\exp\left(-\beta_2 r\right)\right]\ . \tag{1}$$

In our investigation, we consider the hairpin in its environment of a dilute aqueous solution of NaCl. In our model, we propose that this double exponential behavior is associated with two distinguishable charge separated states for the hairpin. *State 1* is the state where Cl^- is attached to Sd^+ and *state 2* is the state where Sd^+ is isolated, that is where Sd^+ only has water molecules nearby. Let Δ denote the energy difference between states 1 and 2, with the phase volume $\Omega >> 1$ for state 2, as Cl^- is dilute in water. Let P_1 and P_2 represent the probabilities of the hairpin occupying states 1 and 2 respectively. Consider time $t = 0$ as the time of hole arrival at Sd.

The time evolution of these two probabilities is determined by two processes including hole recombination with rates k_1 and k_2 from states 1 and 2 respectively and fluctuation of the system between states 1 and 2 with rates w_{12} and w_{21} for transitions $1 \rightarrow 2$ and $2 \rightarrow 1$ respectively. Rates w_{12} and w_{21} characterize the process of counterion dissociation and association respectively. For thermal equilibrium probabilities P_1^* and P_2^*, they must satisfy the detailed balance principle

$$P_1^* w_{12} = P_2^* w_{21}\ . \tag{2}$$

As P_2 is the probability that the Cl^- counterion is not bound to Sd^+, we can express the ratio of probabilities P_2/P_1 as C, where

$$C = \Omega\exp\left(-\Delta/k_B T\right)\ , \tag{3}$$

so that

$$w_{12} = Cw_{21} \ . \tag{4}$$

The time dependence of P_1 and P_2 is determined by the rate equations

$$\begin{aligned} \frac{dP_1}{dt} &= -k_1 P_1 - w_{12} P_1 + w_{21} P_2 \\ \frac{dP_2}{dt} &= -k_2 P_2 - w_{21} P_2 + w_{12} P_1 \ . \end{aligned} \tag{5}$$

An experimentally observable parameter is survival probability $P(t) = P_1(t) + P_2(t)$. Its time decay characterizes the recombination rate.

The solution to (5) is

$$\begin{pmatrix} P_1(t) \\ P_2(t) \end{pmatrix} = f_1 \begin{pmatrix} 1 \\ a \end{pmatrix} \exp(-\lambda_1 t) + f_2 \begin{pmatrix} 1 \\ b \end{pmatrix} \exp(-\lambda_2 t) \ , \tag{6}$$

where

$$\begin{aligned} a &= w_{12}/(-\lambda_1 + k_2 + w_{21}) \\ b &= w_{12}/(-\lambda_2 + k_2 + w_{21}) \ . \end{aligned} \tag{7}$$

The decay rates λ_1 and λ_2 are the solutions to the characteristic equation for (5) and constants f_1 and f_2 are determined by initial conditions $P_1(0)$ and $P_2(0)$, where the survival probability $P(0) = P_1(0) + P_2(0) = 1$. λ_1, λ_2, f_1 and f_2 are determined to be

$$\begin{aligned} \lambda_1 &= \frac{w_{12} + w_{21} + k_1 + k_2}{2} - \sqrt{\frac{(-w_{12} + w_{21} - k_1 + k_2)^2}{4} + w_{21} w_{12}} \\ \lambda_2 &= \frac{w_{12} + w_{21} + k_1 + k_2}{2} + \sqrt{\frac{(-w_{12} + w_{21} - k_1 + k_2)^2}{4} + w_{21} w_{12}} \end{aligned} \tag{8}$$

and

$$\begin{aligned} f_1 &= -\frac{P_2(0) - aP_1(0)}{a - b} \\ f_2 &= \frac{P_2(0) - bP_1(0)}{a - b} \ . \end{aligned} \tag{9}$$

Now we assume recombination is slow; that is $k_1, k_2 < w_{12} + w_{21}$. Indeed, using the diffusion rate $D \approx 10^{-5}\text{cm}^2/\text{s}$ [9] and assuming that a counterion moves by random jumps of 1Å, one can estimate that during the minimum recombination time of 10 ns the counterion jumps approximately 1000 times, which should be sufficient for the ion to access the bound state near the Sd^+ group as $[\text{Cl}^-] \sim 0.1\text{M}$ (about 1 chloride ion per 500 water molecules). Thus the equilibration rate $w = w_{12} + w_{21}$ should exceed the hole recombination rates.

Using our assumption, we can approximate the decay rates $\lambda_{1,2}$ via Taylor expansion by the lowest order in $k_{1,2}/w$ as

$$\begin{aligned} \lambda_1 &\approx \frac{w_{12}k_2 + w_{21}k_1}{w_{12} + w_{21}} \\ \lambda_2 &\approx w_{12} + w_{21} \ . \end{aligned} \tag{10}$$

With this approximation, the solution of (5) reads

$$\begin{aligned} \begin{pmatrix} P_1(t) \\ P_2(t) \end{pmatrix} = (P_1(0) + P_2(0)) \begin{pmatrix} w_{21}/(w_{12} + w_{21}) \\ w_{12}/(w_{12} + w_{21}) \end{pmatrix} \exp(-\lambda_1 t) \\ + \frac{P_1(0)w_{12} - P_2(0)w_{21}}{w_{12} + w_{21}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \exp(-\lambda_2 t) \ . \end{aligned} \tag{11}$$

It follows that the derived survival probability can be expressed as

$$P(t) = P_1(t) + P_2(t) \approx \exp(-\lambda t) \ . \tag{12}$$

Thus the observed decay rate can be well approximated by

$$\lambda_1 = \frac{w_{12}k_2 + w_{21}k_1}{w_{12} + w_{21}} \ . \tag{13}$$

Taking $\lambda_1 = k$, (1) follows directly from (4) and (10) as the thermodynamic average of the two state recombination rates. β_1 and β_2 are taken to be the experimentally derived values 0.97Å^{-1} and 0.42Å^{-1} respectively [7].

A Hückel model is used to describe charge tunneling from Sd^+ to Sa^- [10],[11]. This model should be reasonably relevant in the tunneling regime despite polaron formation and possible environmental fluctuations which can reduce the energy barrier. We can then determine the electron overlap integral b for adjacent AT base pairs. We assume that the thermal equilibrium of the counterion and the Sd^+ ion is established before recombination begins. This is justified by the large diffusion rate of counterions in water: $D \approx 10^{-5}\text{cm}^2/\text{s}$ (see [9]).

According to our model, (3) is the probability that counterion Cl^- is not bound to the Sd^+ group, where Δ is the binding energy and $k_BT = 0.026\text{eV}$ is the thermal energy at room temperature. Experiments [7] were performed in a 0.1M aqueous solution of NaCl. The prefactor Ω can be estimated as the ratio of water molecules per chloride ion, so that $\Omega \sim 556$. The error of this estimate is assumed to be less than an order of magnitude so we can consider $10^2 < \Omega < 10^3$. Lower and upper boundaries for Ω will be used to estimate the accuracy of our estimate of the binding energy Δ. Solving (3) with the probability C taken from experimental data [7] (see (1)), one finds

$$0.36\text{eV} < \Delta < 0.42\text{eV} \ . \tag{14}$$

This estimate is within the range of typical counterion binding energies [12], [13].

In the next step, a relationship is derived between energies E_1 and E_2 of the Sd^+ linker in the presence and absence of Cl^- respectively, and corresponding

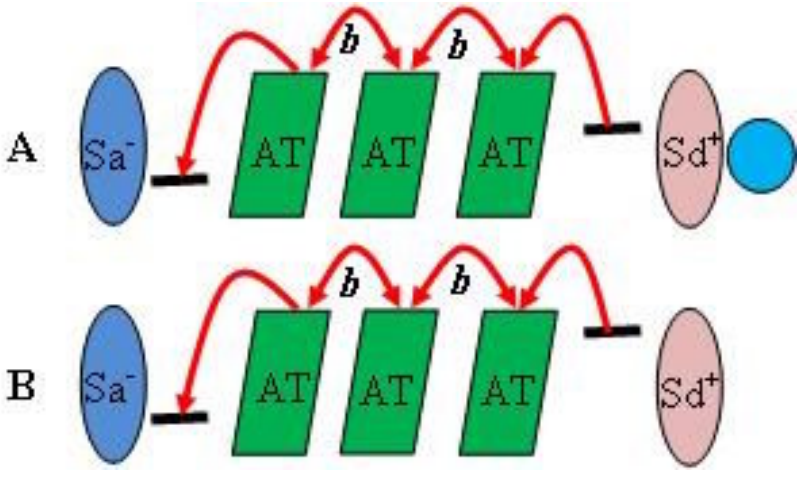


Fig. 2. Stilbene capped hairpin; charge recombination channels

tunneling exponents β_1 and β_2 in (1). The energy of an isolated $(AT)^+$ is set to zero. $E_1 < E_2$ implies that

$$E_2 = E_1 + \Delta \,. \tag{15}$$

One can establish a clear relationship between energy and the tunneling exponent. This relationship involves the electron transfer integral b responsible for charge tunneling between adjacent AT base pairs, shown in Fig.2. It can be written as [10]

$$2b\cosh(\beta_i a/2) = E_i,\ i = 1, 2 \,, \tag{16}$$

where $a = 3.4$Åis the distance between adjacent base pairs.

This relationship can be derived as follows. The tunneling rate through n AT pairs (cf. (1)) is determined by the exponential tail of the positive charge wave function $k_n \sim \psi_n^2$. Here ψ_n^2 is the probability of finding the charge at the nth base pair. The wavefunction of the charge under the barrier decreases exponentially with the base number as $\psi_n \sim \exp(-\kappa n)$, while the charge transfer rate decreases exponentially with the bridge length as $\exp(-\beta a n)$. Comparing the two exponents, we have $\kappa = \beta a/2$. Equation (16) results from the discrete Schrödinger equation [10]

$$E\psi_n = -b(\psi_{n-1} + \psi_{n+1}) \tag{17}$$

if the exponential n dependence for ψ_n is assumed as above ($\psi_n \sim \exp(-\kappa n)$).

Using (14), (15) and (16), one can evaluate the electron transfer integral and the energies of Sd^+ in the absence or presence of Cl^- as $b = 0.13 \pm 0.01$eV, $E_1 = 0.73 \pm 0.06$eV and $E_2 = 0.34 \pm 0.03$eV. It is remarkable that the estimate for the electron transfer integral b agrees very well with the calculations of Voityuk and coworkers [14] for the coupling of adjacent thymine bases, but differs from the estimate for intrastrand coupling of adjacent adenine bases by a factor of 3.

In principle, based on the ionization potentials of DNA bases calculated in a vacuum [14,15,16], one should expect that the hole tunnels through the adjacent adenine bases rather than the thymine bases because the adenine ionization potential is lower by about 1eV. However, the real situation is complicated by the dissolution of DNA in water, which can affect ionization potential. Furthermore, tunneling is also very sensitive to bridge fluctuations [17]. The analysis of other experimental data for charge transfer through AT bridges of various lengths [18] can also be successful when only assuming electron transfer integrals to exceed *ab initio* estimates for them.

It is not quite clear to us as to why there is no observation of thermally activated recombination for $(AT)_n$ bridges up to $n = 7$ in contrast to Ref.[18]. A possible explanation of this behavior is edge effect, which does not affect tunneling, but can be crucially important for thermally activated transport. For instance, the energy of the AT^+ state of an AT pair adjacent to an Sd linker can be larger than the energy of other AT pairs due to its electrostatic interaction with Sd. This difference can increase the activation energy of the first hopping step thus suppressing the hopping channel.

Another reason can be the difference of charge recombination [7] with charge shift reactions [15]. For example, the investigation of charge recombination in DNA hairpins using naphthaldimide and phenothiazine as acceptor and donor separated by various AT bridges [19] exhibits behavior similar to Ref.[7]. Indeed, a tunneling exponent $\beta = 0.40\text{Å}^{-1}$ was reported for charge recombination across 4-8 AT base pairs. The small preexponential factor [19] ($10^8 s^{-1}$) compared to k_0 in (1) can be due to the fact that charge recombination for 4-8 base pairs also occurs without associated counterions.

3 Discussion

Our model can be verified in a number of ways. One way is to change the NaCl concentration. According to (1), the reduction of NaCl concentration by a factor of ten will increase the recombination rate tenfold for long AT bridges ($n > 4$).

A more interesting experiment is to replace chloride ions with other counterions, for instance F^-, Br^- or I^-. The binding energy should decrease with increasing the ionic radius. Crude estimates in Table 1 were made assuming that the binding energy Δ_X of counterion X is inversely proportional to ionic radius. This estimate ignores the contribution of water to binding energy and thus underestimates possible changes in recombination rates for different counterions. We used the same values for the electron transfer integral b, energy E_2 of the Sd^+ state in the absence of a counterion (and, consequently, exponent β_2) and the phase volume factor Ω. The last column in Table 1 shows the recombination rate for the longest bridge of seven AT pairs. For the smallest ion F^-, the recombination time is as long as 3.33 s.

In order to substantiate our hypothesis about counterion attachment to the Sd group and the ionic radius effect on counterion binding energy, we performed very preliminary calculations of binding energies for various halide anions with

Table 1. Approximate parameters of recombination rates controlled by different counterions (see Eq.(1)), $\beta_2 = 0.42\text{Å}^{-1}$

X	DFT Energy (eV)	Δ_X (eV)	β_1 (Å^{-1})	C	k_7 (s^{-1})
F	7.55	0.56	1.14	2.71×10^{-7}	0.3
Cl	4.08	0.41	1.03	8.02×10^{-5}	88
Br	4.42	0.38	1.01	2.68×10^{-4}	293
I	3.88	0.34	0.97	1.31×10^{-3}	1436

the Sd^+ group as shown in Table 1. Calculations were performed in a vacuum; therefore, all energies are overestimated. Calculations in an aqueous solution are currently in progress. The calculations were performed using density functional theory and the B3LYP/3-21G basis set. It is obvious that energies differ from our expectations by a factor of 10 which is a consequence of the absence of Coulomb interactions with water. However, the data trend has a reasonable correlation with our expectation. Moreover, the modeled binding energy of F^- ion is greater than for Cl^- as compared to our expectations for these ions which leads us to expect that the charge recombination rate should be even slower than we predicted compared to the present case of Cl^- ions.

It may seem odd that the binding energy for Br^- is slightly larger than that for Cl^-. This anomaly in the trend of decreasing binding energy of the system with increasing ionic radius of the halide counterions can possibly be understood when considering their electron affinities. We see that the electron affinity of the chloride ion is larger than for the other halide ions. The partial charge in chloride, approximately -0.405C, is twice as high than for other halides; therefore, it has a minimum covalent bond strength which reduces the binding energy compared to our expectations. A similar trend was reported in [20].

4 Conclusion

It is shown that the complicated double exponential recombination kinetics in DNA hairpins can be interpreted assuming that this process is controlled by the binding of a counterion. Key experiments are suggested to verify our theory and to control the recombination process by varying counterions.

Acknowledgments. This work is supported by the NSF CRC Program Grant No. 0628092. GSB acknowledges the support of the Department of Defense Science, Mathematics and Research for Transformation (SMART) Scholarship Program. Authors acknowledge Russ Schmehl for fruitful discussions.

References

1. Tinoco Jr., I.: Hypochromism in polynucleotides. J. Am. Chem. Soc. 82, 4785–4790 (1960)
2. Braun, E., Eichen, Y., Sivan, U., Ben-Yoseph, G.: DNA-templated assembly and electrode attachment of a conducting silver wire. Nature 391, 775–778 (1998)

3. Meggers, E., Michel-Beyerle, M.E., Giese, B.: Sequence Dependent Long Range Hole Transport in DNA. J. Am. Chem. Soc. 120, 12950–12955 (1998)
4. Henderson, P.T., Jones, D., Hampikian, G., Kan, Y., Schuster, G.B.: Long-distance charge transport in duplex DNA: The phonon-assisted polaron-like hopping mechanism. Proc. Natl. Acad. Sci. U.S.A. 96, 8353–8358 (1999)
5. Steckl, A.J.: DNA - a new material for photonics. Nature Photonics 1, 3 (2007)
6. Lewis, F.D., Wu, T., Zhang, Y., Letsinger, R.L., Greenfield, S.R., Wasielewski, M.R.: Distance-Dependent Electron Transfer in DNA Hairpins. Science 277, 673–676 (1997)
7. Lewis, F.D., Zhu, H., Daublain, P., Fiebig, T., Raytchev, M., Wang, Q., Shafirovich, V.: Crossover from superexchange to hopping as the mechanism for photoinduced charge transfer in DNA hairpin conjugates. J. Am. Chem. Soc. 128, 791–800 (2006)
8. Granstrom, M., Petritsch, K., Arias, A.C., Lux, A., Andersson, M.R., Friend, R.H.: Laminated fabrication of polymeric photovoltaic diodes. Nature 395, 257–260 (1998)
9. Berg, R.: Random Walks in Biology. Princeton University Press, Princeton (1983)
10. Berlin, Y.A., Burin, A.L., Ratner, M.A.: On the Long Range Charge Transfer in DNA. Chem. Phys. 275, 61–74 (2002)
11. Berlin, Y.A., Burin, A.L., Ratner, M.A.: Charge Hopping in DNA. J. Am. Chem. Soc. 123, 260–268 (2001)
12. Angelini, T.E., Liang, H., Wriggers, W., Wong, G.: Like-charge attraction between polyelectrolytes induced by counterion charge density waves. Proc. Natl. Acad. Sci. 100, 8634–8637 (2003)
13. Naghizadeh, J.: Counterion binding in polyelectrolyte theory. In: Bennemann, K.H., Brouers, F., Quitmann, D. (eds.) Euro-Par 1982. LNP, vol. 172, pp. 242–246. Springer, New York (1982)
14. Voityuk, A.A., Rösch, N., Bixon, M., Jortner, J.: Electronic coupling for charge transfer and transport in DNA. J. Phys. Chem. 104, 9740–9745 (2000)
15. Siriwong, K., Voityuk, A.A., Newton, M.D., Rösch, N.J.: Estimate of the Reorganization Energy for Charge Transfer in DNA. Phys. Chem. B 107, 2595–2601 (2003)
16. Sugiyama, H., Saito, I.: Theoretical Studies of GG-Specific Photocleavage of DNA via Electron Transfer: Significant Lowering of Ionization Potential and 5'-Localization of HOMO of Stacked GG Bases in B-Form DNA. J. Am. Chem. Soc. 118, 7063–7068 (1996)
17. Troisi, A., Orlandi, G.: The hole transfer in DNA: calculation of electron coupling between close bases. Chem. Phys. Lett. 344, 509–518 (2001)
18. Giese, B., Amaudrut, J., Köhler, A.K., Spormann, M., Wessely, S.: Direct observation of hole transfer through DNA by hopping between adenine bases and by tunnelling. Nature 412, 318–320 (2001)
19. Takada, T., Kawai, K., Cai, X., Sugimoto, A., Fujitsuka, M., Majima, T.: Charge Separation in DNA via Consecutive Adenine Hopping. J. Am. Chem. Soc. 126, 1125–1129 (2004)
20. Ríos, H., Gamboa, C., Ternero, G.: Counterion binding to cationic polyelectrolytes in aqueous solution. Journal of Polymer Science B 29, 805–809 (1991)

Quantum Oscillator in a Heat Bath

Pramodh Vallurpalli, Praveen K. Pandey, and Bhalachandra L. Tembe

Department of Chemistry, I.I.T. Bombay, Mumbai – 400076, India
bltembe@chem.iitb.ac.in

Abstract. In the present article, we use the density matrix evolution method to study the effect of a model solvent on the vibrational spectrum of a diatomic solute particle. The effect of the solvent is considered as a perturbation on the Hamiltonian of the quantum subsystem consisting of a harmonic oscillator. The bath particles are treated classically. The perturbation potential representing the interaction between the solute and the solvent is represented in a bi-exponential form. This provides an effective way to evaluate the required matrix elements needed to compute the evolution of the density matrix. The model calculations indicate that the repulsive parts of the potential dominate causing blue shifts in the vibrational frequencies.

Keywords: Density matrix evolution, vibrational spectrum, quantum harmonic oscillator, bi-exponential perturbation potential, Hellmann-Feynman force, Runge Kutta method, molecular dynamics, bath particles.

1 Introduction

A study of the effects of solvents on the electronic and vibrational structure and dynamics of molecules has interested chemists for a long time. In recent years, the effect of solvation dynamics on the dynamical behaviour of molecules has been studied experimentally as well as theoretically. In theoretical approaches, one often separates the small molecular system from the bulk or the solvent and is able to treat the two subsystems using methods most appropriate to represent the physical conditions. It is very common to treat the molecular system quantum mechanically and the solvent classically and suitably treat the interaction between the two [1-5]. Several approaches have been used to address the problem, such as the wave packet propagation method, the Car Parrinello method, the path integral method and the density matrix evolution (DME) method. While each method has its advantages, we use the density matrix evolution method in the present work. Our specific interest is in the effect of the solvent structure on vibrational spectrum of a diatomic oscillator. The red and blue shifts in the spectrum have been of significant interest for quite some time [6, 7]. An effective numerical method to obtain the time dependence of the density matrix has been developed by Berendsen and coworkers [8-10]. The method is outlined in the next section. The results are presented in Section 3 followed by conclusions in section 4.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 197–202, 2009.

2 The DME Method and the Model Potential

In the DME method, the Liouville-von Neumann equation for the density matrix is numerically solved to obtain the elements of the density matrix as a function of time.

The time-dependent wavefunction of the quantum sub-system is expanded in terms of a suitably chosen orthogonal set of *M* basis functions ϕ_n .

$$\Psi(\xi,t) = \sum_{n=1}^{M} c_n(t)\,\phi_n(\xi) \tag{1}$$

The coordinates of the quantum subsystem are denoted by ξ and the time evolution of the system is studied in terms of the time dependent coefficients $c_n(t)$. In Equation (1), M is the number of basis functions of the harmonic oscillator and we have taken the first 5 harmonic oscillator wavefunctions in our study. The elements of the *M X M* hermitian density matrix ρ_{nm} are defined by

$$\rho_{nm} = c_n\, c_m^{*} \tag{2}$$

The diagonal elements of the density matrix give the populations of the levels and the off diagonal elements contain the phase information. The Liouville- von Neumann equation for the density matrix is given by

$$\frac{d\rho}{dt} = \frac{i}{\hbar}(\rho H - H\rho) \tag{3}$$

In the absence of the solvent, the Hamiltonian is H^0. The presence of the solvent perturbs the system and the perturbed Hamiltonian is given by

$$H = H^0 + H' \tag{4}$$

The unperturbed matrix elements can be evaluated analytically while the perturbed matrix elements have to be evaluated by appropriate expansion of the potential. The effect of the solvent particles on the quantum subsystem is incorporated through H' in Equation (4). The total matrix element is obtained by summing over all the N particles of the solvent.

$$H'_{n,m} = \sum_{i=1}^{N} \langle n\,|H'_i|\,m\rangle \tag{5}$$

Here, H'_i represents the interaction between the i th classical particle and the quantum subsystem [8-10].

The dynamics of the solvent is governed by the classical equations of motion and the quantum subsystem contributes an additional term to the forces on the solvent particles through the Hellmann-Feynman force [11] term which is given by

$$F_{nm,u} = \langle n\,|-\frac{\partial H}{\partial u}|\,m\rangle \tag{6}$$

where *u* represents the directions *x*, *y* and *z* respectively. The total force acting on the classical particle due to the quantum subsystem F_u^Q is

$$F_u^Q = Tr(\rho F_u) \tag{7}$$

Here, Tr represents the trace. Similarly, the vibrational energy of the quantum subsystem is given by

$$E^Q = Tr(\rho H) \tag{8}$$

The total force on the classical particle is given by

$$F_u = F_u^Q - \sum_{j \neq i}^{N} \frac{\partial V_{ij}}{\partial u} \tag{9}$$

For the quantum subsystem, we choose a harmonic oscillator with the potential,

$$V(\xi) = \frac{1}{2} k \xi^2 \tag{10}$$

where $\xi = x_1 - x_2$.

The oscillator is kept fixed on the *x* axis with the two particles of the oscillator located at x_1 and x_2 respectively and is surrounded by an atomic solvent wherein the particles interact via a Lennard-Jones (LJ) potential.

$$V(r) = 4\varepsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^{6} \right] \tag{11}$$

The interaction between the solvent particles and the atoms constituting the oscillator are represented by a bi-exponential function

$$V(r) = A\,\mathrm{e}^{-br} + Ce^{-dr} \tag{12}$$

The values of A, b, C and d and the graph of the above two potentials are given below. This is necessary because, if we use the LJ potential in Equation (6), the matrix elements will diverge since the integration in Equation (6) includes the small values of r as r goes to zero. The matrix elements for the force in Equation (6) are obtaining by expanding the perturbation potential *V* (r) in terms of the vibrational coordinate ξ and the coefficients which depend on the distance between the location of the oscillator atoms and the solvent particles [8-10]. The integrations of the equations of motion for the density matrix are done by using the fourth order Runge Kutta method [12].

3 Results and Discussion

We treat the quantum subsystem as a diatomic oscillator oscillating with the frequency of Cl_2. The two chlorine atoms are placed on the *x*-axis of a periodic box of length 16 Å containing solvent particles interacting with an LJ potential with σ = 3.4 Å and ε / k = 120 K. The Cl atoms and the solvent particles also interact with a similar

potential, except that this potential is expressed in the form of Eq. (12) with the parameters A = 1.8025 x 10^{10} J/mol, b = 5.0081 Å, C = - 25875.225 J/mol and d = 0.8513 Å which preserve the important features of the LJ potential given in (11).

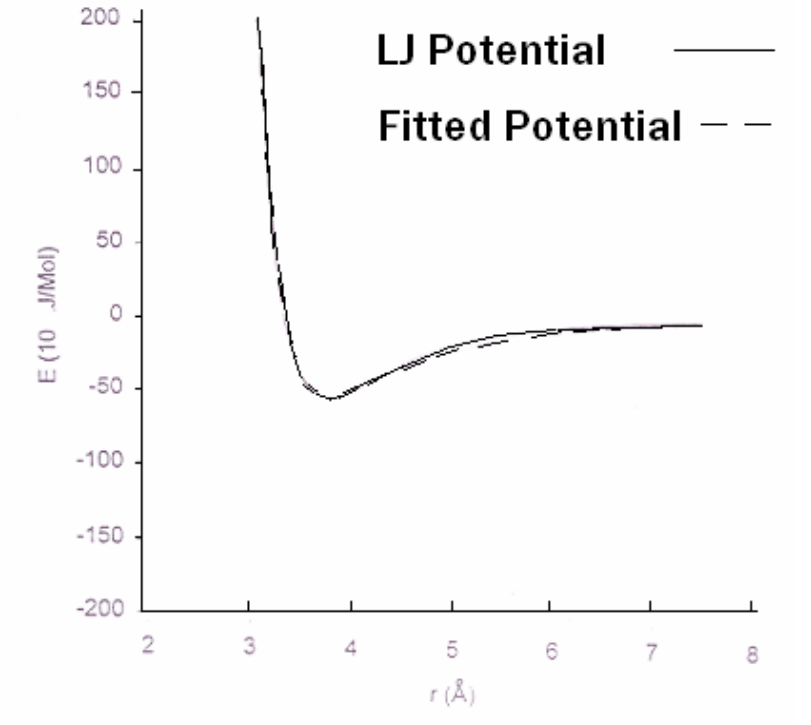


Fig. 1. The Lennard-Jones potential and the fitted bi-exponential function

Table 1. The frequencies of diatomic Cl_2 at various solvent densities. The reduced density is $\rho\sigma^3$ is shown in the first row. The oscillator frequencies at two solvent temperatures are shown below the corresponding densities.

Reduced Density $\rho\sigma^3$	0.154	0.308	0.616	0.770	0.925
Frequencies (cm^{-1}) Cl_2 (500K)	559.0	559.5	560.5	561.5	562.5
Frequencies (cm^{-1}) Cl_2 (1000K)	559.0	56 0.5	561.5	563.5	565.5

The expectation value of ξ is calculated at each time step and its Fourier transform of this expectation value is obtained to obtain the vibrational frequency in the solvent. The production runs extend up to 100 ps with a time step of 0.1fs. We also compute the solute solvent and the solvent solvent radial distribution functions, but these are not reported here as our main interest is in the frequency shifts of the diatomic. The frequencies at various densities of the solvent are given in Table 1. Higher solvent temperatures are chosen so that during the simulations, different elements of the density matrix get significantly populated. At room temperature, the populations of all the elements of the density matrix except ρ_{00} and ρ_{11} do not develop significant populations [13, 14].

4 Conclusions

We have calculated in the present article the vibrational frequencies of a diatomic in the presence of a model solvent at various densities. The method used herein implies an implicit separation between the vibrational motion of the oscillator and the translational motion of the solvent. The model LJ potential between the solvent and the atoms of the oscillator was written as a sum of two exponentials so that the matrix elements required to evaluate the time evolution could be easily computed. We observe that in our simulations at various solvent densities, there is a predominant blue shift in the oscillator frequencies. As has been observed by several earlier investigators, the major components that contribute to the frequency shifts are the attractive and repulsive parts of the potential, the solvent density and the changes in the equilibrium bond length of the oscillator as a function of solvent density. In earlier calculations [6, 7], red shifts have been observed at low densities of the solvent. At higher densities, there was a blue shift in the spectrum. We observe that when the potential well is made deeper by increasing the well depth, there are red shifts in the spectrum in the low density regions. When the frequency of the oscillator was increased (to represent molecules such as N_2), the solvent effects on the oscillator frequency are larger. Red shifts also appear at lower solvent densities. We think that it should be possible to obtain the full density dependence of the vibrational spectrum without using a model potential in which the equilibrium bond length of the quantum oscillator is density dependent [6]. Work is in progress in this direction.

Acknowledgments. We thank I.I.T. Bombay for computational support. We also thank Sridhar Bale, Kanti Prabha, Partha Pratim Das and B. Nivedita for fruitful discussions.

References

1. Kapral, R.: Progress in the Theory of Mixed Quantum-Classical Dynamics. Ann. Rev. Phys. Chem. 57, 129–157 (2006)
2. Berne, B.J., Thirumalai, D.: On the Simulation of quantum systems by Path Integral Methods. Ann. Rev. Phys. Chem. 37, 401–424 (1986)
3. Beck, M.H., Jäckle, A., Worth, G., Meyer, H.-D.: The Multiconfiguration Time-Dependent Hartree (MCTDH) Method: a Highly Efficient Algorithm for Propagating Wavepackets. Physics Reports 324, 1–145 (2000)

4. Car, R., Parrinello, M.: Unified Approach for Molecular Dynamics and Density-Functional Theory. Phys. Rev. Lett. 55, 2471–2474 (1985)
5. Selloni, A., Carnevali, P., Car, R., Parrinello, M.: Localization, Hopping and Diffusion of Electrons in Molten Salts. Phys. Rev. Lett. 59, 823–827 (1987)
6. Herman, M.F., Berne, B.J.: Monte Carlo Simulation of Solvent Effects on Vibrational and Electronic Spectra. J. Chem. Phys. 78, 4103–4125 (1983)
7. de Souza, L.E.S., Guerin, C.B.E., Ben-Amotz, B., Szleifer, I.: Statistical mechanics of solvent induced forces and vibrational frequency shifts. Low density expansions and Monte Carlo simulations. J. Chem. Phys. 99, 9954–9961 (1993)
8. Berendsen, J.H.C., Mavri, J.: Quantum Simulation of Reaction Dynamics by Density Matrix Evolution. J. Phys. Chem. 97, 13464–13468 (1993)
9. Mavri, J., Berendsen, J.H.C.: Dynamical Simulation of a Quantum Harmonic Oscillator in a Noble Gas by Density Matrix Evolution. Phys. Rev. E. 50, 198–204 (1994)
10. Mavri, J., Berendsen, J.H.C.: Calculation of the Proton Transfer Rate Using Density Matrix Evolution and Molecular Dynamics Simulations: Inclusion of the Proton Excited States. J. Phys. Chem. 99, 12711–12717 (1995)
11. Feynman, R.: Forces in Molecules. Phys. Rev. 56, 340–343 (1939)
12. Kreyszig, E.: Advanced Engineering Mathematics, 9th edn. John Wiley and Sons, Chichester (2005)
13. Pramodh, V.: Ab initio MD applied to chemical systems. M. Sc. Thesis, Department of Chemistry, I.I.T. Bombay (1997)
14. Pandey, P. K.: Density Matrix Evolution Study for Diatomics. M. Sc. Thesis, Department of Chemistry, I.I.T. Bombay (1999)

Density Functional Theory Study of Ag-Cluster/CO Interactions

Paulo H. Acioli, Narin Ratanavade, Michael R. Cline, and Sudha Srinivas

Department of Physics and Astronomy
Northeastern Illinois University
Chicago, IL, 60625
{p-acioli,s-srinivas}@neiu.edu
http://physics.neiu.edu/∼acioli
http://physics.neiu.edu/∼srinivas

Abstract. The interactions between carbon monoxide and small clusters of silver atoms are examined. Optimal geometries of the cluster-molecules complexes, i.e. silver cluster - carbon monoxide molecule, are obtained for different sizes of silver clusters and different numbers of carbon monoxide molecules. This analysis is performed in terms of different binding energy of these complexes and analysis of the frontier orbitals of the complex compared to those of its constituents. The silver atom and the dimer (Ag_2) bond up to three carbon monoxide molecules per Ag atom, while the larger clusters appear to saturate at two CO's per Ag atom. Analysis of the binding energy of each CO molecule to the cluster reveals that the general trend is a decrease with the number of CO molecules, with the exception of Ag where the second CO molecule is the strongest bound. A careful analysis of the frontier orbitals shows that the bent structures of AgCO and Ag_2CO are a result from the interaction of the highest occupied orbital of Ag (5s) and Ag_2 (σ) with the lowest unoccupied orbital of CO (π^*). The same bent structure also appears in the bonding of CO to some of the atoms in the larger clusters. Another general trend is that the CO molecules have a tendency to bond atop of an atom rather than on bridge or face sites. These results can help us elucidate the catalytic properties of small silver clusters at the atomic level.

1 Introduction

Studies directed at understanding the interaction of metal clusters with molecules remain the subject of active theoretical and experimental studies [1]. The recent shift in the focus of enquiry from the study of the structural and electronic properties of single-component metal clusters to the study of the interaction of metal clusters with molecules is driven in large part by the potential use of metal clusters in heterogeneous catalysis. Clusters have been long known to show size-specific physical and chemical properties that are often at odds with their bulk properties. This size-specific behavior is evident in the catalytic

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 203–210, 2009.

properties as well, leading to highly reactive clusters whose catalytic behavior is at odds with their bulk counterparts. For example, it has been known for some years now that gold nanoclusters supported on metal oxides such as TiO_2 can efficiently oxidize carbon dioxide at fairly low temperatures [2]. More recently, the eight atom gold cluster supported on an MgO substrate has been shown to be the smallest clusters capable of catalyzing the oxidation of carbon monoxide [3].

The situation for silver clusters is less conclusive. In the bulk and the near bulk regime of several micrometers, silver plays a unique role that is not replicated by any other transition metal in the selective oxidation of ethylene into ethylene oxide, a widely-used industrial catalytic process [4]. Yet, little is known about the details of the mechanism of the oxidation reaction at an atomistic level. There is significant progress of late in experimental techniques that can be used to soft-land clusters on to substrates to perform precise size specific investigations to study the size-specific chemical reactivity of clusters and consequently their catalytic properties. Likewise, advances in first-principles theoretical methods such as density functional theory and quantum chemistry techniques provide another direct route to studying the chemical properties of the clusters at an atomistic level. Since it is as yet unclear as to whether silver clusters exhibit the catalytic oxidation of CO like gold clusters do, it is perhaps not surprising that several theoretical [5,6,7,8,9] and experimental [9,10] investigations have attempted to address the interaction of silver clusters with O_2 and CO in an attempt to study the fundamental cluster-molecule interaction as the first step towards understanding the larger question of the role, if any, of silver clusters in the catalytic oxidation of CO.

In this paper, we examine the interaction of small silver clusters (Ag_n, $1 \leq n \leq 4$) with one or more CO molecules. The ground (2S) state of the Ag atom is well-separated first excited (2D) state, the d orbitals are localized relative to the 5s orbitals and bonding in the silver clusters is dominated by the 5s electrons. It is not surprising therefore that silver clusters display structural and electronic properties that are quite similar to those of alkali metal clusters. The geometric and electronic properties of pure silver clusters have been well studied [11,12,13,14] and the candidates for the lowest energy structures in the size range of $n \leq 10$ that is of interest to us are well-elucidated. We study the interaction of the cluster molecules system under the framework of the generalized gradient approximation of the density functional theory. The functionals used in the calculations presented in this paper combine Becke exchange functional [15] with the Perdew-Wang [16] correlation functional as implemented inGaussian 03 [17]. A 28-electron core ([Ar]3d10) effective core potential is combined with contracted Gaussian-type orbitals for the remaining 19 electrons for the Ag atoms [18]. Carbon and oxygen atoms are represented by the DGTZVP[19] all-electron basis sets as implemented in Gaussian 03 [17]. The selection of pseudopotential and exchange-correlation functionals was made to compare with previous results on the bare clusters as reported in ref. [14] and the choice in that reference was based in tests to reproduce the properties of Ag, Ag_2 and Ag_3. The DGTZVP was chosen to reproduce the experimental binding energy of CO. The

equilibrium structures were obtained using gradient-based methods. The optimizations were performed in all degrees of freedom. Normal mode analysis was performed to determine whether a structure was a minimum or a saddle point in the potential energy surface of the cluster. The starting point was the bare silver cluster structures of ref. [14]. CO molecules were added until saturation was reached. The saturation was 3 CO molecules per Ag atom for the Ag and Ag_2. Ag_3 and Ag_4 bonded 2 CO molecules per silver atom. The bonding of CO to the cluster is analyzed in terms of the binding energy of CO to the clusters.

2 Results

2.1 Bare Silver Clusters

In this subsection we briefly review the properties of small silver clusters as obtained in the previous study of Srinivas *et al.* [14]. In Fig. 1 we present the lowest energy isomers of Ag_n, n =1 -4, and their respective binding energies as obtained in ref. [14] and confirmed by our current results. Ag_4 is a rhombus rather than a tetrahedron which is a common form for tetramers of other elements. A 3-dimensional form will only form for neutral silver clusters containing 7 or more atoms[14]. The binding energy in the range of n$\leq$ 10 increases, non-monotonically, with the number of atoms in the cluster. In this work we will restrict ourselves to the smaller size clusters (n$\leq$4) as we are more interested in understanding the binding mechanism of carbon monoxide to the bare and CO-rich silver clusters.

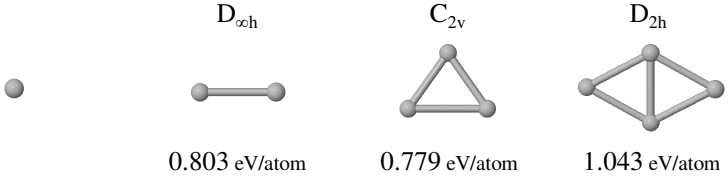


Fig. 1. Structure of the most stable isomer of Ag_n, $n = 1 - 4$

2.2 $Ag_N(CO)_x$

To understand the mechanism and the energetics of the bonding of CO to small silver clusters we begin with the structures presented in Fig. 1 and add CO molecules until they no longer bond to the clusters. This determination is made when the binding energy of the last CO to the previous cluster is negative or zero. In Fig. 2 we present the structure of $Ag(CO)_x$, $x = 1-3$. One can see that in both AgCO and $Ag(CO)_2$ the CO molecule is bent with respect to the Ag-C bond. A careful analysis of the frontier orbitals of both the Ag atom and dimer and of CO shows that the π^* orbital of CO tends to align with the 5s orbital of Ag or the σ orbital of Ag_2, resulting in the bent structure of Fig. 2. In the following analysis we will consider the binding energy of the CO to the complex $Ag(CO)_{n-1}$ as

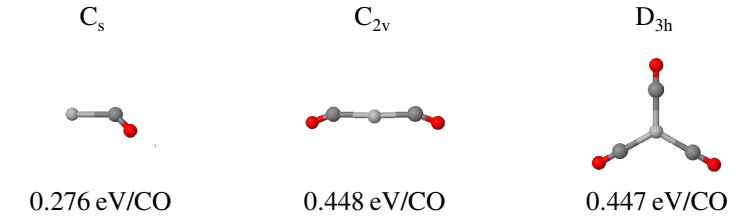


Fig. 2. Structure of $Ag(CO)_x$, $x = 1-3$. For each structure we present the point group symmetry and the corresponding binding energy per CO ligand. The silver atoms are light grey, the C atoms dark grey and the O are red.

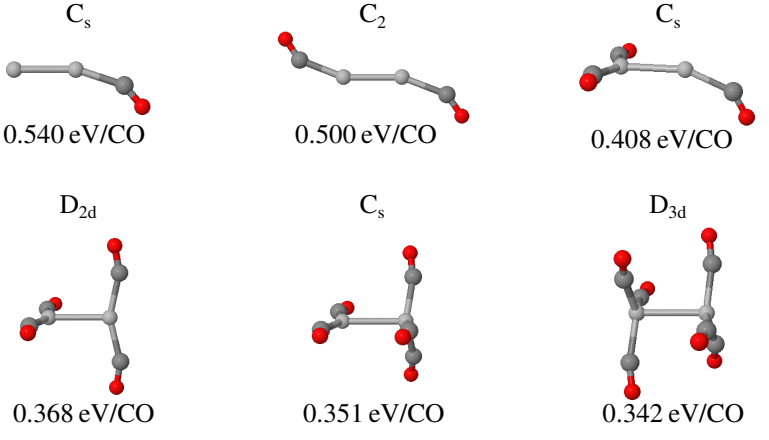


Fig. 3. Structure of $Ag_2(CO)_x$, $x = 1-6$. For each structure we present the point group symmetry and the corresponding binding energy per CO ligand.

BE(last CO) = E(CO)+E($Ag(CO)_{n-1}$)-E($Ag(CO)_n$). Using this energy we can determine the relative stability of the complex as CO molecules are added. The binding energy of CO to Ag is relatively weak, for the first CO it is about 0.276 eV, it increases for the second CO to 0.619 eV, and then decreases again to 0.446 eV when a third carbon monoxide is added. This explains why the binding energy per CO defined as BE(CO) = [n×E(CO)+E(Ag(CO))-E($Ag(CO)_n$)]/n reported in Fig. 2 for $Ag(CO)_2$ and $Ag(CO)_3$ are nearly the same. A fourth CO did not bond to the cluster. $Ag(CO)_3$ is very symmetric, belonging to the D_3h point symmetry group. This suggests, that although the third CO is weakly bonded, each Ag atom can bond up to 3 CO's.

In Fig. 3 we display the $Ag_2(CO)_x$, $x = 1-6$, indicating again that although a weakly bonded system, each Ag atom can bond up to 3 carbon monoxide molecules. Just as discussed in the previous paragraph, in all the cases studied for Ag_2 the O atom is not in the same line as the Ag-C bond. A careful analysis of the frontier orbitals once again show us that this is a result of the interaction of the π^* orbital of CO with the orbitals of Ag. The cluster becomes less stable

upon addition of CO. However, the smallest binding energy of CO to Ag_2 is still comparable with the binding of a single CO to the Ag atom.

The structures of the $Ag_3(CO)_x$, $x = 1 - 6$ are displayed in Fig. 4. The first thing to note is that in $Ag_3(CO)$ and $Ag_3(CO)_3$ the top CO is in perfect alignment with the Ag-C bond. In this case the orbitals that interact are the π^* of CO with a π-like orbital of Ag_3. However, this trend is not followed for all single CO bonded to Ag, as one can see in the case of $Ag_3(CO)_2$. Ag_3 bonds up to 2 CO's per Ag atom, something that can be explained by the extra Ag-Ag bond. A possibility of binding 3 CO's per Ag atoms would be as a linear Ag_3 backbone and each $Ag(CO)_3$ stacked in a staggered fashion, as a sandwich cluster. This structure demonstrated to be unstable and we don't think larger sizes will be stable. The binding energy per CO molecule shows a decreasing trend, but like in the case of Ag_2, the strength of the weakest CO-Ag bond in $Ag_3(CO)_6$ is comparable to the Ag-CO bond in AgCO. In $Ag_3(CO)_5$ the 2 CO's bonded to a single Ag atom are perpendicular to the plane of the molecule. While in $Ag_3(CO)_6$ the last two CO's are in the same plane as Ag_3. One can reason that these are steric effects that will tend to minimize the repulsion between the CO molecules.

The last set of structures studied in this work is displayed in Fig. 5. Both Ag_4CO and $Ag_4(CO)_2$ have the CO molecule aligned with the Ag-CO bond. The explanation is very similar to the one for the case of Ag_3. But, due to the higher symmetry of Ag_4 the second CO bonds in the same fashion as the first. Each Ag atom will bond a single CO atoms before a second CO will bond to it. This was the trend for the smaller sizes as well. $Ag_4(CO)_8$ has a very similar structure to $Ag_3(CO)_6$, it consists of two pair of CO molecules bonded in the plane perpendicular to Ag_4 and two pairs that are parallel to its plane. In the cases explored in this work not a single CO bonded in a face or a bridge site, a fact that can attributed to both the frontier orbitals of CO and Ag_n.

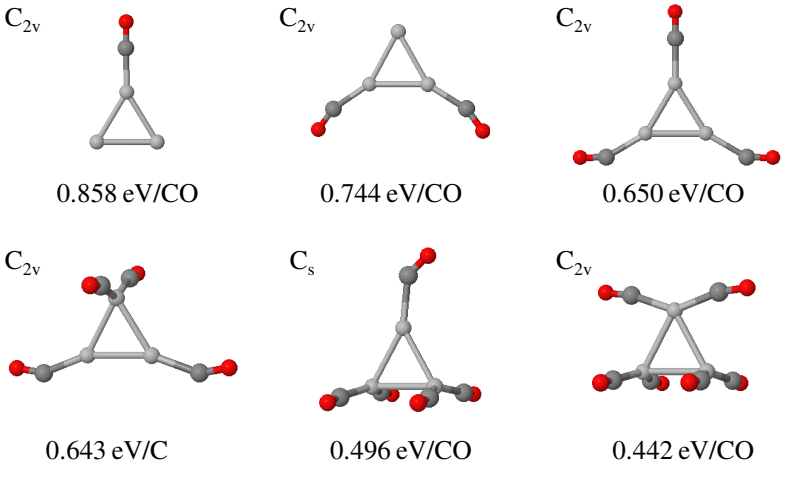


Fig. 4. Structure of $Ag_3(CO)_x$, $x = 1 - 6$. For each structure we present the point group symmetry and the corresponding binding energy per CO ligand.

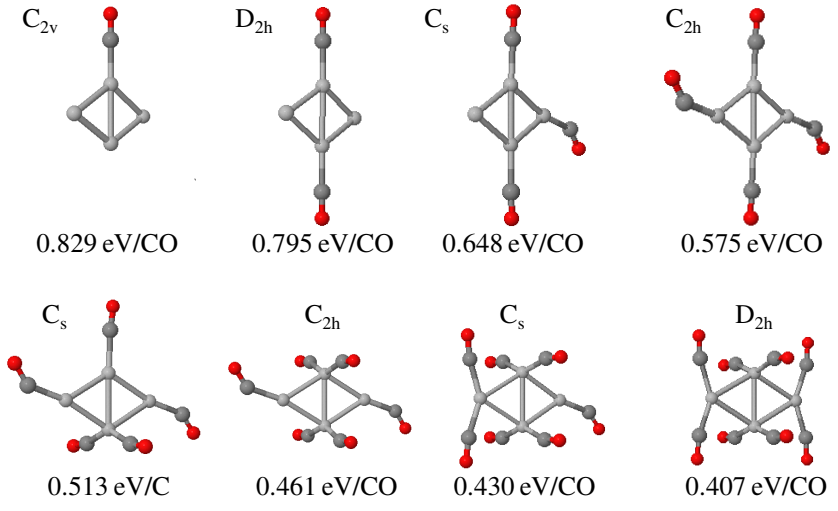


Fig. 5. Structure of $Ag_4(CO)_x$, x= 1 − 8. For each structure we present the point group symmetry and the corresponding binding energy per CO ligand.

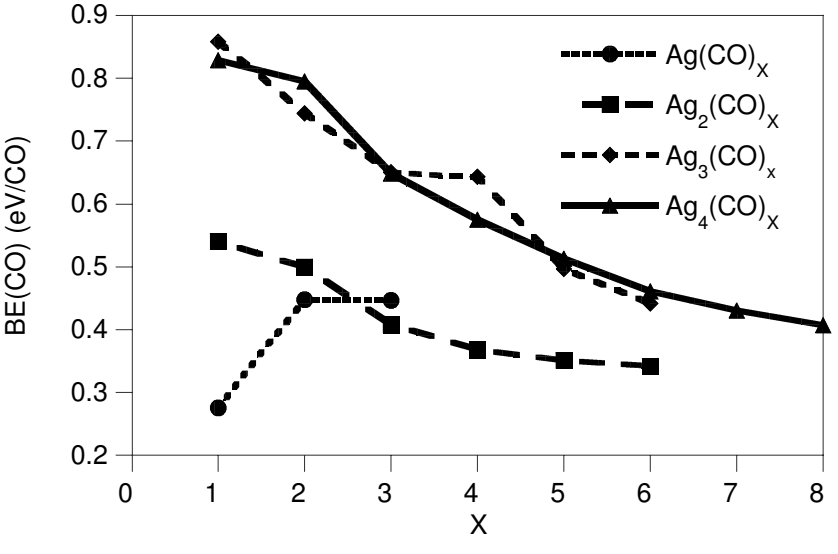


Fig. 6. The binding energy of CO (per CO ligand) to Ag_N, $N = 1 - 4$. The binding energy is computed as $BE(\mathrm{CO}) = (x \times E(\mathrm{CO}) + E(\mathrm{Ag}_N) - E(\mathrm{Ag}_N(\mathrm{CO})_x)/x$.

In Fig. 6 we present the binding energy per CO to Ag_n, $n = 1-4$ as a function of the number of CO molecules bonded to the cluster. This quantity is a measure of the stability of the cluster upon addition of ligand molecules. The first thing we learn from Fig. 6 is that the binding energy of CO to Ag_n increases with

cluster size. However, as more CO molecules are added the binding energy for all the sizes seem to decrease to approximately the same value. The data in the figure also allow us to conclude that the only case in which the system becomes more stable upon addition of more CO molecules is the Ag atom. This might be understood in terms of the orbitals that are responsible to the formation of the carbon-silver bond. Ag_2, Ag_3, and Ag_4 all become less stable upon addition of carbon monoxide. As a consequence at high temperatures one might only observe $Ag_N CO$. Nevertheless, in low temperature conditions or in clusters deposited on a substrate there is a possibility that these CO-rich clusters might form and play a role in the reactions involving carbon monoxide.

3 Concluding Remarks

In this work we studied the structural properties and the interaction of carbon monoxide with silver clusters. We conclude that low coordination Ag atoms can bond up to 3 CO's, being the last CO molecule only weakly bonded. Analysis of the frontier orbitals explains why in many cases that CO molecule is not aligned with the Ag-C bond. The binding energy per CO molecule decreases with the number of CO molecules bonded to the cluster, the only exception is the Ag atom where the second CO molecule is more strongly bonded than the first. Steric effects explain the structure of the saturated clusters. $Ag_2(CO)_6$ consists of two $Ag(CO)_3$ structures staggered in a D_3d arrangement. While $Ag_3(CO)_6$ has two pairs of CO molecules bonded perpendicular to the plane of Ag_3 and the 3rd pair is parallel. A similar situation is seen in $Ag_4(CO)_8$ where two pairs are perpendicular and two pairs are parallel to the plane of Ag_4.

We speculate that the likely saturation of the planar Ag_5 and Ag_6 clusters to be 10 and 12 pairs of CO molecules bonded in patterns similar to the ones for Ag_3 and Ag_4. On the other hand, for the tridimensional structures the saturation point may be a single CO molecule per surface atom due to their higher coordination. Future work will be to extend this work to larger size clusters and to explore the electronic properties of neutral and charged clusters. In addition, we would like to explore other ligands and study catalytic properties of small silver clusters in the gas phase as well as of silver clusters deposited on a substrate.

Acknowledgments

We would like to acknowledge the financial support from a NEIU COR grant.

References

1. Ervin, K.M.: Metal-ligand interactions: gas-phase transition metal cluster carbonyls. Int. Rev. Phys. Chem. 20, 127–164 (2001); Bernhardt, T. M.: Gas-phase kinetics and catalytic reactions of small silver and gold clusters. Int. Jour. Mass. Spec. 243, 1-29 (2005); and references therein
2. Valden, M., Lai, X., Goodman, D.W.: Onset of Catalytic Activity of Gold Clusters on Titania with the Appearance of Nonmetallic Properties. Science 281, 1647–1650 (1998)

3. Yoon, B., Häkkinen, H., Landman, U., Wörz, A., Antonietti, J.-M., Abbet, S., Judai, K., Heiz, U.: Charging Effects on Bonding and Catalyzed Oxidation of CO on Au8 Clusters on MgO. Science 307, 403–407 (2005)
4. Carter, E.A., Goddard III, W.A.: Chemisorption of oxygen, chlorine, hydrogen, hydroxide, and ethylene on silver clusters: A model for the olefin epoxidation reaction. Surf. Sci. 209, 243–289 (1989)
5. Boussard, P.J.E., Seigbahn, P.E.M., Svensson, M.: The interaction of ammonia, carbonyl, ethylene and water with the copper and silver dimers. Chem. Phys. Lett. 231, 337–344 (1994)
6. Zhou, J., Li, Z.-H., Wang, W.-N., Fan, K.-N.: Density functional study of the interaction of carbon monoxide with small neutral and charged silver clusters. J. Phys. Chem. A 110, 7167–7172 (2006)
7. Jiang, L., Xu, Q.: Infrared Spectra of the $(AgCO)_2$ and Ag_nCO (n = 2-4) Molecules in Rare-Gas Matrices. J. Phys. Chem. A 110, 11488–11493 (2006)
8. Giordano, L., Vitto, A.D., Pachionni, F., Ferrari, A.M.: CO adsorption on Rh, Pd and Ag atoms deposited on the MgO surface: a comparative ab initio study. Surf. Sci. 540, 63–75 (2003)
9. Bernhardt, T.M., Socaciu-Siebert, L.D., Hagen, J., Wöste, L.: Size and composition dependence in CO oxidation reaction on small free gold, silver, and binary silver-gold cluster anions. Appl. Catal. A 291, 170–178 (2005)
10. Hagen, J., Socaciu-Siebert, L.D., Le Roux, J., Popolan, D., Vajda, S., Bernhardt, T.M., Wöste, L.: Charge transfer initiated nitroxyl chemistry on free silver clusters Ag^-_{2-5}: Size effects and magic complexes. Intl. J. Mass. Spectr. 261, 152–158 (2007)
11. Bonacic-Koutecky, V., Cespiva, L., Fantucci, P., Koutecky, J.: Effective core potential-configuration interaction study of electronic structure and geometry of small neutral and cationic Ag_n clusters: Predictions and interpretation of measured properties. J. Chem. Phys. 98, 7981–7994 (1993)
12. Kaplan, I.G., Santamaria, R., Novaro, O.: Theoretical-study of the geometric structures and energetic properties of anionic clusters - Ag^-_n (n=2 to 6). Int. J. Quant. Chem. Symp. 27, 743–753 (1993)
13. Poteau, R., Heully, J.-L., Spiegelmann, F.: Structure, stability, and vibrational properties of small silver cluster. Z. Phys. D. 40, 479–482 (1997)
14. Srinivas, S., Salian, U., Jellinek, J.: Theoretical Investigations of Silver Clusters and Silver-Ligand Systems. In: Russo, M., Salahub, D.R. (eds.) Metal-ligand interactions in chemistry, physics, and Biology. Kluwer Academic Publishers, Dordrecht (2000)
15. Becke, A.D.: Density-functional exchange-energy approximation with correct asymptotic-behavior. Phys. Rev. A 38, 3098–3100 (1988)
16. Perdew, J.P., Wang, Y.: Accurate and simple analytic representation of the electron-gas correlation-energy. Phys. Rev. B 45, 13244–13249 (1992)
17. Frisch, M.J., et al.: Gaussian 2003. Gaussian, Inc., Wallingford (2004)
18. Andrae, D., Haussermann, U., Dolg, M., Stoll, H., Preuss, H.: Energy-adjusted ab-initio pseudopotentials for the 2nd and 3rd row transition-elements. Theor. Chim. Acta 77, 123–141 (1990)
19. Godbout, N., Salahub, D.R., Andzelm, J., Wimmer, E.: Optimization of Gaussian-type basis-sets for local spin-density functional calculations. 1. boron through neon, optimization technique and validation. Can. J. Chem. 70, 560–571 (1992); Sosa, C., Andzelm, J., Elkin, B. C., Wimmer, E., Dobbs, K. D., Dixon, D. A.: A local density functional-study of the structure and vibrational frequencies of molecular transition-metal compounds. J. Phys. Chem. 96 , 6630–6636 (1992)

Time-Dependent Density Functional Theory Study of Structure-Property Relationships in Diarylethene Photochromic Compounds

Pansy D. Patel[1,2] and Artëm E. Masunov[1,2,3]

[1] NanoScience Technology Center
[2] Department of Chemistry
[3] Department of Physics, 12424 Research Parkway, Suite 400, University of Central Florida, Orlando, FL 32826 USA
amasunov@mail.ucf.edu

Abstract. Photochromic compounds exhibit reversible transition between closed and open isomeric forms upon irradiation accompanied by change in their color. The two isomeric forms differ not only in absorption spectra, but also in various physical and chemical properties and find applications as optical switching and data storage materials. In this contribution we apply Density Functional Theory (DFT) and Time-Dependent DFT (TD-DFT) to predict the equilibrium geometry and absorption spectra of a benchmark set of diarylethene based photochromic compounds in open and closed forms (before and after photocyclization). Comparison of the calculated Bond Length Alternation parameters with those available from the X-ray data indicates M05-2x functional to be the best method for geometry optimization when basis set includes polarization functions. We found M05 functional accurately predicts the maximum absorption wavelength when solvent is taken into account. We recommend combined theory level TD-M05/6-31G*/PCM//M05-2x/6-31G*/PCM for prediction of geometrical and spectral parameters of diarylethene derivatives.

Keywords: photochromism, density functional theory, electronic spectra, bond length alternation, molecular structure.

1 Introduction

Photochromism is light-induced reversible molecular transition between two isomers, closed and open, with different absorption spectra. Apart from the color, the two isomers also differ in various physical and chemical properties such as refractive indices, dielectric constants, oxidation-reduction potentials and geometrical structures. The instant property changes upon photoirradiation can be used in various optoelectronic devices such as optical memory, optical switching, displays and nonlinear optics. Irie and Lehn [1-9] were among the first authors to investigate diarylethenes as a potential candidate for photochromic applications (Fig.1).

In the case of photochromic diarylethenes, the open form has twisted π-system and is colorless while the closed form with nearly planar π-system is conjugated and colored.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 211–220, 2009.

Thus, the ground state geometry is essential to predict their characteristic properties. An important geometrical parameter in the conjugated systems is the bond-length alternation (BLA), defined as the difference between the single and double bond lengths. For linear chain oligomers it has been known that the band gap, nonlinear optical (NLO) properties, excited states, etc. are BLA-dependent [10-15].

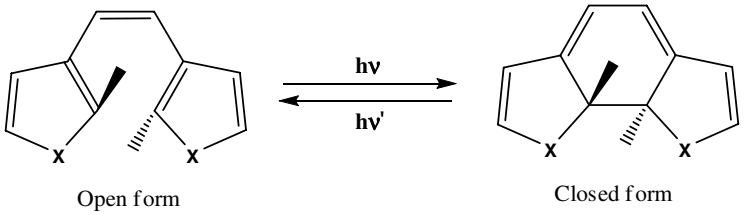


Fig. 1. Photochromic diarylethene compounds (X=S,O,Se)

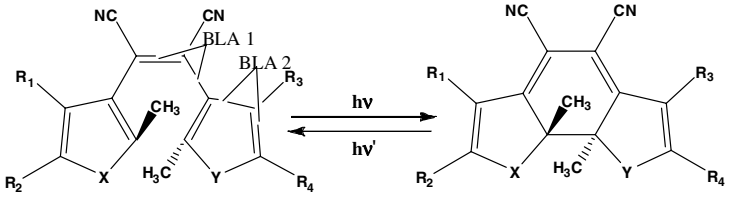
The theoretical predictions of BLA for several series of conjugated oligomers has been conducted by Jacquemin and co-workers [16-23] in the past decade. They performed *ab initio* calculations on mainly acyclic conjugated systems and concluded that (1) MP2 values are in good agreement with higher-order electron-correlated wavefunction approaches that include triple excitations; (2) basis set effects are relatively limited, and polarized double-ζ basis is sufficient, at least for DFT calculations; (3) all conventional GGA and meta-GGA provide similar BLA, that are much too small and too rapidly decreasing with the chain lengthens; (4) hybrid functionals correct this trends but to a small extend so that quantitative agreement with MP2 values is still far away; (5) the conformation differences do not alter these three latter conclusions; (6) self-interaction corrections included via the averaged-density self-interaction correction (ADSIC) scheme improves BLA evolution obtained by the conventional DFT approaches. For medium-size oligomers ADSIC predicts BLA in better agreement with MP2, than B3LYP or PBE0. However, diarylethene derivatives had not been investigated in that respect.

In the present contribution we report BLA using different DFT methods to predict the ground state geometry for the open and closed isomers as well as for some by-products. The methods are validated by comparison with the experimental X-ray crystal structures available for some of diarylethene derivatives. Our goal is to establish the computational protocol to investigate structure-property relationships for the diarylethene derivatives aimed to guide the design of new photochromics.

The distinctive absorption spectrum of the two isomeric forms of the photochromic compounds is an essential property of investigation. Experimental absorption spectra (λ_{max}) of such compounds are determined in different solvents for different derivatives. Recently Jacquemin and co-workers evaluated the λ_{max} for large set of perfluoro derivatives of diarylethenes solvent conditions using Time-Dependent Density Functional (TD-DFT) formalism [24]. However their data is limited for closed isomers only. In the present paper, we have employed TD-DFT formalism to predict the absorption spectra of a benchmark set of photochromic compounds for both open and closed isomeric forms.

2 Computational Details

The calculations have been performed using GAUSSIAN03 package. Different levels of theory were used to find the best method for geometry optimization, followed by absorption spectra predictions. Complete optimizations have been performed on a benchmark set of diarylethene photochromic compounds (Fig.2,a-d) to perform bond length alternation (BLA) analysis and compared to the experimentally determined X-ray geometries of a set of structures in order to validate a suitable method as well as basis set for accurate geometry prediction.



a.

Comp	X	Y	R1	R2	R3	R4
DCN-1	S	N-CH_3	CH_3	CH_3	CH=CH-CH=CH	
DCN-2	N-CH_3	N-CH_3	CH_3	CH_3	CH=CH-CH=CH	

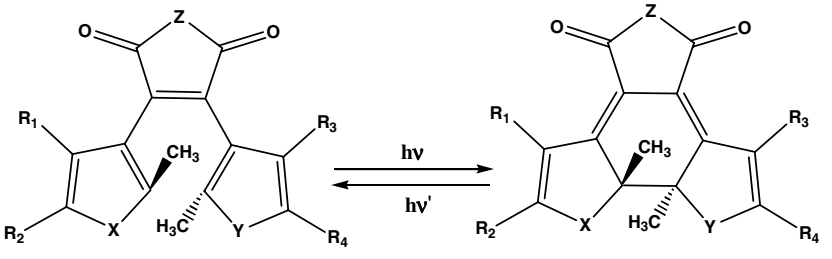
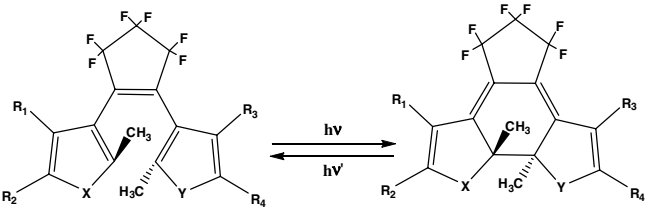


Figure:2b

b.

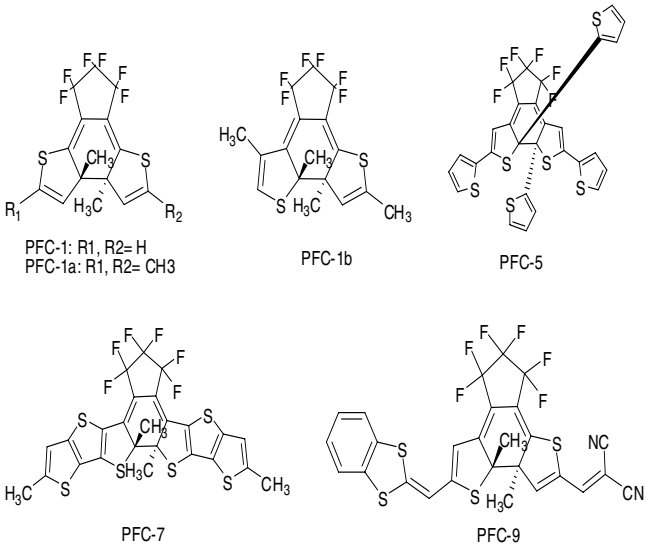
Comp	X	Y	Z	R1	R2	R3	R4
MA-1	S	S	O	CH_3	CH_3	CH_3	CH_3
MA-1-A	S	S	O	CH=CH-CH=CH		CH=CH-CH=CH	
MA-2	S	N-CH_3	O	CH_3	CH_3	CH=CH-CH=CH	
MA-2-A	S	N-CH_3	O	CH_3	CH_3	CH=CH-CH=CH	
MA-2-B	S	N-CH_3	O	CH_3	CN	CH=CH-CH=CH	
MA-3	N-CH_3	N-CH_3	O	CH=CH-CH=CH		CH=CH-CH=CH	
MA-hit	S	S	O	H	CH_3	H	CH_3
Mi	S	S	NH	CH_3	CH_3	CH_3	CH_3

Fig. 2(a,b). Benchmark set of open and closed-ring isomers studied in this work (DCN-Dicyano derivatives, MA-Maleicanhydride derivatives and Mi- Maleimide derivatives)



c.

Comp	X	Y	R1	R2	R3	R4
PFC-1-C	S	S	CH_3	H	CH_3	H
PFC-1-D	S	S	H	CH_3	H	CH_3
PFC-1-E	S	S	CH_3	CH_3	CH_3	CH_3
PFC-2	S	S	H	Ph	H	Ph
PFC-2-A	S	S	CH_3	Ph	CH_3	Ph
PFC-2-B	S	S	CH_3	Ph-N-C_2H_5	CH_3	Ph-N-C_2H_5
PFC-3	S	S	H	Th	H	Th
PFC-4	S	S	H	Th-CH_3	H	Th-CH_3
PFC-6	S	N-CH_3	CH=CH-CH=CH		CH=CH-CH=CH	
PFC-6-A	S	N-CH_3	CH_3	CN	CH=CH-CH=CH	
PFC-8	S	S	CH=CH-CH=CH		CH=CH-CH=CH	
PFC-B	O	O	CH_3	H	CH_3	H



d.

Fig. 2(c,d). (PFC- Perfluorocyclopentene derivatives)

The optimized structures were further used to predict the excitation spectrum of each molecule with Time-Dependent DFT (TD-DFT) formalism. TD-DFT is a quantum mechanical method used to investigate the excited state proprieties of many-body systems. It is important to note that out of several excited states only the one with the maximal oscillator strength was used for comparison with experiment. Often that was

not the lowest excitation reported by TD-DFT. Several different functionals have been tested to select the best method which can be used to determine the accurate absorption spectra for both isomeric forms of the different derivatives of diarylethenes. Solvent effects were included implicitly by means of non-equilibrium polarizable continuum model (PCM), which uses empirical dielectric constants (both slow orientational and fast electronic components) as well as atomic radii as model parameters. PCM typically provides a good approximation of solvent effects as long as specific interaction with the solvent (such as hydrogen bonds) can be neglected.

The solvents used for the current work were chosen to reproduce the experimental results as close as possible. Heptane (Hep) was used for the compounds whose experimental data was available in hexane while Benzene (Bz), Dicholomethane (DCM) and Acetonitrile (ACN) was used for those compounds whose experimental data was available in the same solvent.

3 Results and Discussions

We conduced the geometry optimization at DFT theory level with various exchange-correlation potentials, including B3LYP, BLYP, BHandHLYP, PBE0, TPSS, BMK,

Table 1. Bond length alternation (BLA, Å) and wavelength of the maxima on the absorption spectra (λmax, nm) for a set of diarylethenes calculated at TD-M05/6-31G*/PCM//M052x/6-31G*/PCM theory level and compared to the experimental data. See Fig.2a for definition of BLA1 and BLA2.

	Closed isomer			**Open isomer**		
	BLA1	**BLA2**	**λ_{max}**	**BLA1**	**BLA2**	**λ_{max}**
PFC-1-d						
Experiment[a]	0.095	0.091	505	-0.112	0.089	303
Theory	0.106	0.087	505	-0.113	0.080	316
PFC-1-e						
Experiment[b]			529	-0.132	0.095	
Theory	0.113	0.093	526	-0.117	0.089	279
PFC-2						
Experiment[c]	0.085	0.055	575	-0.112	0.050	276
Theory	0.100	0.076	585	-0.114	0.068	287
PFC-2-et						
Experiment[d]	0.089	0.059	600	-0.115	0.068	286
Theory	0.101	0.075	611	-0.116	0.067	284
PFC-B						
Experiment[e]	0.113	0.055	469	-0.120	0.053	274
Theory	0.119	0.045	476	-0.102	0.057	251
PFC-5						
Experiment[f]			632	-0.133	0.062	320
Theory	0.101	0.071	611	-0.116	0.060	332
MA-hit-closed						
Experiment[g]			510	-0.109	0.082	403
Theory	0.091	0.077	520	-0.102	0.071	423
RMSD	0.006	0.007	4	0.004	0.003	6

Ref - a-[25], b-[26], c-[27], d-[28],e-[29], f-[30], g-[31]

Table 2. Maximum absorption wavelengths (λ_{max}, nm) measured experimentally and predicted at two theory levels: TD-M05/6-31G*/PCM (**T1**) andTD-B3LYP/6-31G*/PCM (**T2**), both use geometry optimized at M052x/6-31G*/PCM level for open and closed isomers of diarylethenes in solution. Deviations of the theoretical values from the experimental ones ($\Delta\lambda_{max}$, nm) are also reported.

		Closed					Open				
		λ			Δλ		λ			Δλ	
Molecule	Solvent	Exp	**T1**	**T2**	**T1**	**T2**	Exp	**T1**	**T2**	**T1**	**T2**
DCN-1[a]	**Bz**	547	552	531	-5	16	412	457	433	-45	-21
DCN-2[a]	**Bz**	574	556	533	18	41	390	480	377	-90	13
MA-1[b]	**Bz**	560	525	531	35	29	335	397	380	-62	-45
MA-1-A[c]	**Bz**	544	538	531	6	13	417	504	475	-87	-58
MA-2[a]	**Bz**	595	563	545	32	50	450	507	481	-57	-31
MA-2-A[d]	**Bz**	680	683	644	-3	36	-	498	493	-	-
MA-2-B[d]	**Bz**	628	624	598	4	30	-	504	481	-	-
MA-3[e]	**Bz**	620	595	565	25	55	470	540	508	-70	-38
Mi[f]	**Bz**	512	496	500	16	12	370	391	374	-21	-4
MA-hit[g]	**Bz**	510	519	520	-9	-10	403	446	423	-43	-20
PFC-1[h]	**Hep**	432	428	436	4	-4	316	342	332	-26	-16
PFC-1-a[i]	**Hep**	425	421	425	4	0	336	357	345	-21	-9
PFC-1-b[i]	**Hep**	469	462	466	7	3	312	311	322	1	-10
PFC-1-c[j]	**Hep**	534	522	528	12	6	234	288	280	-54	-46
PFC-1-d[i]	**Hep**	505	499	505	6	0	303	326	316	-23	-13
PFC-1-e[k]	**Hep**	529	517	505	12	24	-	285	266	-	-
PFC-2[l]	**Hep**	575	590	585	-15	-10	280	298	287	-18	-7
PFC-2-a[j]	**Hep**	562	576	575	-14	-13	262	294	280	-32	-18
PFC-2-b[j]	**Hep**	597	602	593	-5	4	305	324	308	-19	-3
PFC-2-et[m]	**Hep**	600	613	611	-13	-11	286	303	288	-17	-2
PFC-3[n]	**ACN**	605	610	604	-5	1	312	315	304	-3	8
PFC-4[n]	**ACN**	612	619	610	-7	2	320	321	312	-1	8
PFC-5[o]	**DCM**	632	629	611	3	21	320	356	332	-36	-12
PFC-6[p]	**ACN**	565	552	534	13	31	340	368	356	-28	-16
PFC-6-A[d]	**Hep**	665	653	625	12	40	-	375	355	-	-
PFC-7[p]	**ACN**	612	596	585	16	27	290	334	327	-44	-37
PFC-8[q]	**Hep**	517	523	521	-6	-4	258	269	261	-11	-3
PFC-9[r]	**Bz**	828	787	792	41	36	354	379	358	-25	-4
PFC-B[s]	**Hep**	469	491	476	-22	-7	274	258	251	16	23
RMSD					3	4				7	4

Ref: a-[9], b-[8], c-[32], d-[7], e-[33], f-[34], g-[31], h-[35], i-[36], j-[37], k-[26], l-[27], m-[28], n-[38], o-[30], p-[39], q-[40], r-[6], s-[29].

M05, and M05-2x. The results of these calculations (which will be published elsewhere) suggest that the M05-2x functional that includes 52% fraction of the Hartree-Fock exchange, gives the best agreement with the experimental BLA values. We also compared the maxima on the absorption spectra, evaluated using TD-DFT formalism with the same selection of exchange-correlation potentials using implicit solvent model for both closed and open isomers. We found that M05 method agrees with the experimental λ_{max} values the best. Polarizable continuum model and double-ζ basis set

with polarization functions were important to obtain the accurate equilibrium geometry as well as absorption spectra. The comparison of the calculated and experimental BLA parameters and absorption wavelengths for the benchmark subset of diarylethene photochromic compounds is reported in Table 1.

For the rest of the molecules in the benchmark set single crystal X-ray diffraction data were not available. We report their maximum absorption wavelengths at two theory levels: TD-M05/6-31G*/PCM and TD-B3LYP/6-31G*/PCM (with geometry optimized at M052x/6-31G*/PCM level) and compare our predictions with the experimental λ_{max} values in Table 2. Looking at the RMSD values reported in the last row of that table one can see that B3LYP functional predicts the wavelengths three times closer to experimental values for the closed ring isomers with extended conjugation lengths, than for the open ring isomers. Other functionals, such as BMK, exhibit an opposite trends. The M05 functional seems to be the best compromise, with the average errors of 4-7 nm.

4 Conclusions

Several exchange-correlation functionals in combination with TD-DFT formalism were evaluated for predictions of the absorption spectra for both closed and open isomers of diarylethene photochromic compounds. Bond length alternation descriptors were employed to select suitable DFT methods to predict equilibrium geometry in these compounds. We found that a) the most accurate equilibrium geometry based on BLA parameter is best calculated at M05-2x/6-31G*/PCM level; b) TD-DFT spectral data is best reproduced at M05/6-31G*/PCM level with the average deviation form the observed values in the range of 3-7 nm; c) use of polarization functions in the basis set is important to obtain the best geometry; d) solvent effects as described by polarizable continuum model (PCM) are important for the accurate predictions of the spectral data with TD-DFT. We recommend theory level TD-M05/6-31G*/PCM//M052x/6-31G*/PCM for prediction of geometrical and spectral parameters (BLA and the λ_{max} values) for both closed and open isomers of diarylethene derivatives. This opens a possibility to establish structure-property relationship for diarylethene photochromics to assist in rational design of improved materials for photoswitching and data storage applications.

Acknowledgements

This work was supported in part by the National Science Foundation Grant No. CCF 0740344. The authors are thankful to DOE NERSC, UCF I2Lab, and UCF Institute for Simulations and Training (IST) HPC Stokes facility for the generous donation of the computer time.

References

1. Nakamura, S., Yokojima, S., Uchida, K., Tsujioka, T., Goldberg, A., Murakami, A., Shinoda, K., Mikami, M., Kobayashi, T., Kobatake, S., Matsuda, K., Irie, M.: Theoretical investigation on photochromic diarylethene: A short review. J. Photochem. Photobiol. A-Chem. 200, 10–18 (2008)

2. Yamaguchi, T., Takami, S., Irie, M.: Photochromic properties of 1,2-bis (6-substitute-2-methyl-1-benzofuran-3-yl) ethene derivatives. J. Photochem. Photobiol. A-Chem. 193, 146–152 (2008)
3. Yamaguchi, T., Uchida, K., Irie, M.: Photochromic properties of diarylethene derivatives having benzofuran and benzothiophene rings based on regioisomers. Bull. Chem. Soc. Jpn. 81, 644–652 (2008)
4. Gilat, S.L., Kawai, S.H., Lehn, J.M.: Light-Triggered Electrical and Optical Switching Devices. J. Chem. Soc.-Chem. Commun., 1439–1442 (1993)
5. Gilat, S.L., Kawai, S.H., Lehn, J.M.: Light-Triggered Electrical and Optical Switching Devices. Molecular Crystals and Liquid Crystals Science and Technology Section a-Molecular Crystals and Liquid Crystals 246, 323–326 (1994)
6. Gilat, S.L., Kawai, S.H., Lehn, J.M.: Light-Triggered Molecular Devices - Photochemical Switching of Optical and Electrochemical Properties in Molecular Wire Type Diarylethene Species. Chem.-Eur. J. 1, 275–284 (1995)
7. Irie, M.: Photochromism: Memories and switches - Introduction. Chemical Reviews 100, 1683 (2000)
8. Irie, M., Mohri, M.: Thermally Irreversible Photochromic Systems - Reversible Photocyclization of Diarylethene Derivatives. J. Org. Chem. 53, 803–808 (1988)
9. Nakayama, Y., Hayashi, K., Irie, M.: Thermally Irreversible Phtotchromic Systems - Reversible Photocyclization of Nonsymmetrical Diarylethene Derivatives Bull. Chem. Soc. Jpn. 64, 789–795 (1991)
10. Bartkowiak, W., Zalesny, R., Leszczynski, J.: Relation between bond-length alternation and two-photon absorption of a push-pull conjugated molecules: a quantum-chemical study. Chem. Phys. 287, 103–112 (2003)
11. BlanchardDesce, M., Alain, V., Bedworth, P.V., Marder, S.R., Fort, A., Runser, C., Barzoukas, M., Lebus, S., Wortmann, R.: Large quadratic hyperpolarizabilities with donor-acceptor polyenes exhibiting optimum bond length alternation: Correlation between structure and hyperpolarizability. Chem.-Eur. J. 3, 1091–1104 (1997)
12. Bourhill, G., Bredas, J.L., Cheng, L.T., Marder, S.R., Meyers, F., Perry, J.W., Tiemann, B.G.: Experimental Demonstration of the Dependence of the 1st Hyperpolarizability of Donor-Acceptor-Substituted Polyenes on the Ground-State Polarization and Bond-Length Alternation. J. Am. Chem. Soc. 116, 2619–2620 (1994)
13. Choi, C.H., Kertesz, M., Karpfen, A.: The effects of electron correlation on the degree of bond alternation and electronic structure of oligomers of polyacetylene. J. Chem. Phys. 107, 6712–6721 (1997)
14. Kirtman, B., Champagne, B., Bishop, D.M.: Electric field simulation of substituents in donor-acceptor polyenes: A comparison with ab initio predictions for dipole moments, polarizabilities, and hyperpolarizabilities. J. Am. Chem. Soc. 122, 8007–8012 (2000)
15. Meyers, F., Marder, S.R., Pierce, B.M., Bredas, J.L.: Electric-Field Modulated Nonlinear-Optical Properties of Donor-Acceptor Polyenes - Sum-Over-States Investigation of the Relationship Between Molecular Polarizabilities (Alpha, Beta, and Gamma) and Bond-Length Alternation. J. Am. Chem. Soc. 116, 10703–10714 (1994)
16. Jacquemin, D., Femenias, A., Chermette, H., Ciofini, I., Adamo, C., Andre, J.M., Perpete, E.A.: Assessment of several hybrid DFT functionals for the evaluation of bond length alternation of increasingly long oligomers. J. Phys. Chem. A 110, 5952–5959 (2006)
17. Jacquemin, D., Perpete, E.A.: Ab initio calculations of the colour of closed-ring diarylethenes: TD-DFT estimates for molecular switches. Chem. Phys. Lett. 429, 147–152 (2006)

18. Jacquemin, D., Perpete, E.A., Chermette, H., Ciofini, I., Adamo, C.: Comparison of theoretical approaches for computing the bond length alternation of polymethineimine. Chem. Phys. 332, 79–85 (2007)
19. Jacquemin, D., Perpete, E.A., Ciofini, I., Adamo, C.: Assessment of recently for the evaluation of the developed density functional approaches bond length alternation in polyacetylene. Chem. Phys. Lett. 405, 376–381 (2005)
20. Perpete, E.A., Jacquemin, D.: An ab initio scheme for quantitative predictions of the visible spectra of diarylethenes. J. Photochem. Photobiol. A-Chem. 187, 40–44 (2007)
21. Perpete, E.A., Maurel, F., Jacquemin, D.: TD-DFT investigation of diarylethene dyes with cyclopentene, dihydrothiophene, and dihydropyrrole bridges. J. Phys. Chem. A 111, 5528–5535 (2007)
22. Perrier, A., Maurel, F., Aubard, J.: Theoretical study of the electronic and optical properties of photochromic dithienylethene derivatives connected to small gold clusters. J. Phys. Chem. A 111, 9688–9698 (2007)
23. Perrier, A., Maurel, F., Aubard, J.: Theoretical investigation of the substituent effect on the electronic and optical properties of photochromic dithienylethene derivatives. J. Photochem. Photobiol. A-Chem. 189, 167–176 (2007)
24. Maurel, F., Perrier, A., Perpete, E.A., Jacquemin, D.: A theoretical study of the perfluorodiarylethenes electronic spectra. J. Photochem. Photobiol. A-Chem. 199, 211–223 (2008)
25. Kobatake, S., Yamada, T., Uchida, K., Kato, N., Irie, M.: Photochromism of 1,2-bis(2,5-dimethyl-3-thienyl)perfluorocyclopentene in a single crystalline phase. J. Am. Chem. Soc. 121, 2380–2386 (1999)
26. Yamada, T., Kobatake, S., Irie, M.: Single-crystalline photochromism of diarylethene mixtures. Bull. Chem. Soc. Jpn. 75, 167–173 (2002)
27. Irie, M., Lifka, T., Kobatake, S., Kato, N.: Photochromism of 1,2-bis(2-methyl-5-phenyl-3-thienyl)perfluorocyclopentene in a single-crystalline phase. J. Am. Chem. Soc. 122, 4871–4876 (2000)
28. Kobatake, S., Shibata, K., Uchida, K., Irie, M.: Photochromism of 1,2-bis(2-ethyl-5-phenyl-3-thienyl)perfluorocyclopentene in a single-crystalline phase. Conrotatory thermal cycloreversion of the closed-ring isomer. J. Am. Chem. Soc. 122, 12135–12141 (2000)
29. Yamaguchi, T., Irie, M.: Photochromism of bis(2-alkyl-1-benzofuran-3-yl)perfluorocyclopentene derivatives. J. Org. Chem. 70, 10323–10328 (2005)
30. Peters, A., McDonald, R., Branda, N.R.: Regulating pi-conjugated pathways using a photochromic 1,2-dithienylcyclopentene. Chem. Commun., 2274–2275 (2002)
31. Shirinyan, V.Z., Krayshkin, M.M., Belenkii, L.I.: Photochromic dihetarylethenes. 8. A new approach to the synthesis of 3, 4-bis (2, 5-dimethyl-3-thienyl)furan-2, 5-dione as potential photochrome, 81 (January 2001); Khim. Geterotsiklicheskikh Soedin., 426 (2001)
32. Uchida, K., Nakayama, Y., Irie, M.: Thermally Irreversible Photochromic Systems - Reversible Photocyclization of 1,2-Bis(benzo[b]thiophen-3-yl)ethene Derivatives. Bull. Chem. Soc. Jpn. 63, 1311–1315 (1990)
33. Nakayama, Y., Hayashi, K., Irie, M.: Thermally Irreversible Photochromic Systems - Reversible Photocyclization of 1,2-Diselenenylethene and 1,2-Diindolylethene Derivatives. J. Org. Chem. 55, 2592–2596 (1990)
34. Uchida, K., Kido, Y., Yamaguchi, T., Irie, M.: Thermally irreversible photochromic systems. Reversible photocyclization of 2-(1-benzothiophen-3-yl)-3-(2 or 3-thienyl)maleimide derivatives. Bull. Chem. Soc. Jpn. 71, 1101–1108 (1998)
35. Fukaminato, T., Kawai, T., Kobatake, S., Irie, M.: Fluorescence of photochromic 1,2-bis(3-methyl-2-thienyl)ethene. J. Phys. Chem. B 107, 8372–8377 (2003)

36. Irie, M., Uchida, K., Eriguchi, T., Tsuzuki, H.: Photochromism of Single-Crystalline Diarylethenes. Chem. Lett., 899–900 (1995)
37. Irie, M., Sakemura, K., Okinaka, M., Uchida, K.: Photochromism of dithienylethenes with electron-donating substituents. J. Org. Chem. 60, 8305–8309 (1995)
38. Peters, A., Branda, N.R.: Electrochemically induced ring-closing of photochromic 1,2-dithienylcyclopentenes. Chem. Commun., 954–955 (2003)
39. Moriyama, Y., Matsuda, K., Tanifuji, N., Irie, S., Irie, M.: Electrochemical cyclization/cycloreversion reactions of diarylethenes. Org. Lett. 7, 3315–3318 (2005)
40. Hanazawa, M., Sumiya, R., Horikawa, Y., Irie, M.: Thermally Irreversible Photochromic Systems - Reversible Photocyclization of 1,2-Bis(2-methylbenzo[b]thiophen-3-yl)Perfluorocycloalkene Derivatives. J. Chem. Soc.-Chem. Commun., 206–207 (1992)

Free Energy Correction to Rigid Body Docking : Application to the Colicin E7 and Im7 Complex

Sangwook Wu[1], Vasu Chandrasekaran[1], and Lee G. Pedersen[1,2]

[1]Department of Chemistry
University of North Carolina,
Chapel Hill, NC 27599-3290
[2]Laboratory of Structural Biology,
NIEHS, RTP, NC 27709-12233
sangwoow@email.unc.edu,
vasu@email.unc.edu,
pederse3@niehs.nih.gov

Abstract. We performed a 2-dimensional free energy calculation in the conformational space composed of two structures, best RMSD (Root Mean Square Distance) and the worst RMSD structures using ZDOCK on the Colicin E7 (protein) and Im7 (Inhibitor) complex. The lowest free energy minimum structure is compared to the X-ray crystal structure and the best RMSD docking structure. The free energy correction for the best RMSD structure shows an alternative in the prediction of a flexible loop position, which could not describe rigid body docking.

Keywords: Free energy calculation, docking, Colicin E7-Im7 complex.

1 Introduction

Docking is a method for estimating the near native structure for a protein-protein complex or a protein-ligand complex through shape or chemical (hydrophobic, hydrophilic) complementarity [1,2]. For example, the near native structure for a protein-protein complex can be easily found if a complementary shape exists between the interface of two proteins. For an efficient search for the near native docked structure from a set of mostly "*incorrectly*" docked structures, the docking method performs a rough prediction by treating the two proteins as rigid bodies. Through appropriate rotation and translation of the two *rigid bodies* using fast Fourier Transforms (FFTs) [3], a score is assigned using a *scoring function* that depends on how close two proteins fit at the complementary interface. At this stage of rough prediction, the shape information about two entities plays a dominant role. As well as shape information, a docking method may also make use of complementary information provided by electrostatics or hydrophobic interactions at the interface of the protein-protein complex. The information about the chemical complementarity between two protein complexes leads to an energetic correction to the shape complementary based on the FFT technique,

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 221–228, 2009.

which is adapted by FTDOCK [4]. For a more elaborate correction of the implicit solvent model, a desolvation energy correction using the ACE (Atomic Contact Energy) [5] is added to the FFT technique and the electrostatic correction in ZDOCK [6]. In addition to electrostatics, the geometry-based hydrophobic complementarity at the interface of protein-protein complex has been incorporated into the FFT-based algorithm in MolFit [7]. Recently, more advanced algorithms have been developed to improve the rigid body docking based on FFT through the incorporation of a pairwise structure-based potential in PIPER [8]. At the refinement stage of prediction, more elaborate algorithms such as FlexE [9] are adapted into the docking methods to describe flexible side chains or backbone movements through the superimposed structures of the ensemble. Dock 4.0 [10] has been implemented with an incremental construction and random search algorithm. Molecular Dynamics (MD) simulation in an explicit solvent model has been applied to myosin phosphatase targeting subunit (MYPT) and its binding site in protein phosphatase-1 (PP1) [11] with a 2-5 ns simulation and FK506 (ligand)-FKBP (FK binding protein) with a 1 ns simulation [12]. However, despite these several algorithms, the incorporation of information about flexible side chain and backbone movements remains one of the main challenges facing the modern docking method. In this study, we propose a free energy correction to the rigid body docking method. The free energy method through effective conformational sampling using the WHAM (Weighted Histogram Analysis Method) procedure [13,14,15,16,17] is able to predict the lowest free energy minimum structure of a protein-protein complex. One of the advantage of the free energy technique in the prediction of near native structure for a protein-protein complex is that it incorporates the dynamics of the protein-protein complex. It can provide us with information of flexible loop movements, depending on the timescale of the MD simulation. In addition, it also provides us with entropy information as well as the energetics of the protein-protein complexes. Such entropy information can prevent the overestimation of the energetics involving the residues at the interface of the protein-protein complex. Entropy is a significant factor which strongly influences the formation of a complex. The general free energy method for predicting protein-protein complex requires a huge amount of sampling in conformational space, especially if starting as a blind trial of the conformational sampling. Such a sampling task becomes aggravated as the protein size increases, and such requires expensive computational expenditure. However, use of rigid body docking, a global search for a "nearly correctly docked structure" from "the set of incorrectly docked structures", could dramatically decrease the burden of conformational sampling for the free energy calculation. In this case, the conformational sampling for the free energy calculation is focused on the region of near native structures for the protein-protein complex identified by rigid body docking. Here, we suggest an efficient method for prediction of protein-protein complexes by *combining* the free energy method with the docking method.

2 Methods

2.1 Rigid Body Docking

The free energy method combined with the docking method is applied to a trial protein-inhibitor system : DNAase domain of Colicin E7 (Protein)−Im7 (Inhibitor) for which we have experimental information. Colicins are protein toxins produced by *Escherichia coli* [18]. The cytotoxic activity is known to be suppressed by binding with its inhibitor [19]. One of the interesting features of colicin and its inhibitors is that the binding affinity is among the strongest known in protein-inhibitor interactions [20]. A test case for the free energy docking technique was selected from the decoy set of the published "Protein-Protein Docking Benchmark" [21]. The benchmark includes individually crystallized receptor and ligand PDBs, along with the co-crystallized complex PDBs for testing protein docking algorithms. The endonuclease domain of Colicin E7 in complex with its inhibitor Im7 protein was used as the test case due to the relatively small size of the protein-protein complex. Two structures were chosen for the references for the free energy calculations based on their RMSDs with respect to the crystallized structure (PDB code : 7CEI). The deviations of C_α RMSD values of the docked structures (compared to X-ray crystal structure) generated by ZDOCK is in the range of from 2.12 Å (Best RMSD) to 36.64 Å (Worst RMSD)[1]. The conformations of the two structures, best RMSD and the worst RMSD, is shown in Fig. 1. The other intermediate structures lie between the two extremes that we have chosen.

2.2 Order Parameter and Free Energy Surface

We choose the Q value, the similarity index between two conformations, as an order parameter for the free energy calculation. Q has been widely used in the free energy calculations in protein folding studies [22,23]. It is defined as

$$Q_A = \frac{1}{N} \sum_{ij} \exp\left[-\frac{(r_{ij} - r_{ij}^A)^2}{2\sigma^2} \right] \tag{1}$$

where r_{ij} is the distances between i-th and j-th atom in conformation of interest, r_{ij}^A is the same for the conformation A for which the Q_A value is defined, and normalization factor N is equal to number of pairs of atoms whose positions define the conformation. The similarity index Q_A changes from 1 (for the conformation A) to 0 (for a conformation with no resemblance to A). Normally, only C_α carbons are chosen in the calculation of the Q value. To track the flexible protein movements, however, we extend the range of atoms to include C_β, C_γ, C_δ, C_ε and C_Z atoms [24]. In Eq. 1, σ controls the resolution of the Q value. Considering that the resolution of the X-ray crystal of 7CEI is 2.3 Å [19], σ is set to 2 Å in our study. A total of 878 atoms are considered in the calculation

[1] The RMSD values were calculated after the alignment of the Im7 (Residue 1 to 87) using VMD.

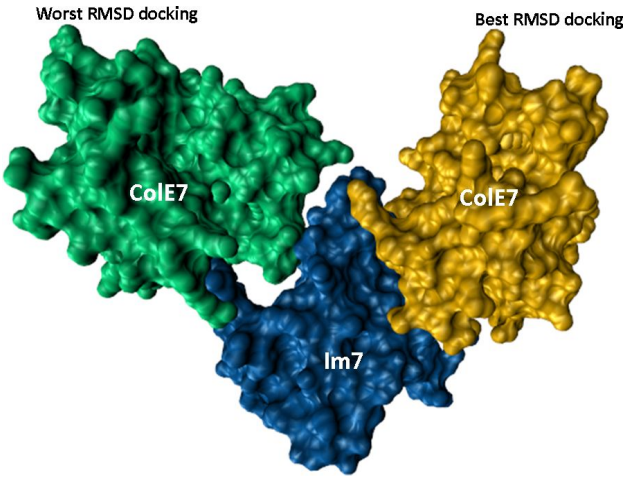


Fig. 1. The best RMSD (Yellow) and worst RMSD (Green) docking structures. The cognate inhibitor, Im7, is in common and shown as a blue color. Yellow represents the DNAase domain of Colicin E7 in the best RMSD structure and green represents the DNAase domain of Colicin E7 in the worst RMSD structure.

of Q_A value; this definition is essential in tracking the conformational change which involve small movements or rotations of flexible side chains. We choose two references, the best docked structure and the worst docked structure as shown in Fig. 1. The basic idea is to perform all the conformational sampling using WHAM between these two extreme references.

2.3 MD Simulation

We performed free energy calculations in the 2-dimensional conformation space composed of ($Q_{bestRMSD}$, $Q_{worstRMSD}$). The conformational sampling is guided by a biasing potential,

$$V(Q_{best}, Q_{worst}) = \frac{1}{2}k_{best}(Q_{best} - Q_{best}^{\min})^2 + \frac{1}{2}k_{worst}(Q_{worst} - Q_{worst}^{\min})^2 \quad (2)$$

where k_{best} and k_{worst} are spring constants and $Q_{best}^{\min}$ and $Q_{worst}^{\min}$ are the locations at which the biasing potentials are applied. The spring constants k_{best} and k_{worst} are in the range of from 11.5 kcal/mol/Å^2 to 82.5 kcal/mol/Å^2. These spring constants are determined so as to obtain the best overlap between trajectories for good sampling. For dielectric constant ϵ=80.0, a total of 367 windows are used for each different $Q_{worst}^{\min}$ and $Q_{best}^{\min}$ ranging from 0.5 to 1. Each window was run for 50 ps. For productive data, the first 10 ps simulation is removed. Thus, the total sampling corresponds to 14.7 ns (40 ps × 367 windows). On the other hand, for dielectric constant ϵ=4.0, a total of 388 windows are used for each different $Q_{worst}^{\min}$ and $Q_{best}^{\min}$ ranging from 0.5 to 1. Each window was run for 100 ps. For productive data, the first 10 ps simulation is removed. Thus, the

total sampling corresponds to 32.8 ns (90 ps × 364 windows). All of the MD simulations were performed using LAMMPS (Large-scale Atomic and Molecular Massively Parallel Simulator) at the atomistic level with the CHARMM27 protein-lipid force field [25]. The best RMSD structure and the worst RMSD structures were treated with a dielectric constant of 80.0 and 4.0 using the distance dependent dielectric solvent model, $\epsilon(r)=\epsilon r$. The two references, the best RMSD and the worst RMSD structures, were minimized using the steepest descent gradient method in NAMD employing the CHARMM force field. The **charm2lammps** perl script [25] converted each minimized structures into the initial structures for LAMMPS. These were equilibrated for 1 ns at 293K. In the process of equilibration at 293 K, target Molecular Dynamics was performed for the two structures to keep the two structures less than 0.1 Å of the backbone RMSD with respect to the two references. The Coulombic and Lennard-Jones interactions were calculated with a 10.0/12.0 Å twin-range cutoff. This is a feasibility study; the validity of the method is to be established (below). The various computational compromises can be removed in future work.

3 Result

Figure 2 shows that the 2-dimensional free energy surface (FES). (Q_{best},Q_{worst}:1.0, 0.59) and (Q_{best},Q_{worst}:0.59, 1.0) corresponds to the best RMSD and the worst RMSD structures respectively. It is gratifying that the lowest free energy structure at ϵ=4.0 is quite similar to the X-ray crystal structure with C_α RMSD of 2.05 $\mathring{A}$. Figure 3.a shows the superimposed images of best RMSD (Cyan) and the lowest free energy minimum structure (Orange). Figure 3.b shows the superimposed images of the the lowest free energy minimum structure (Orange) and the X-ray crystal (Green). The main difference at the backbone level between the lowest free energy structure and X-ray structure lies in loop configurations from

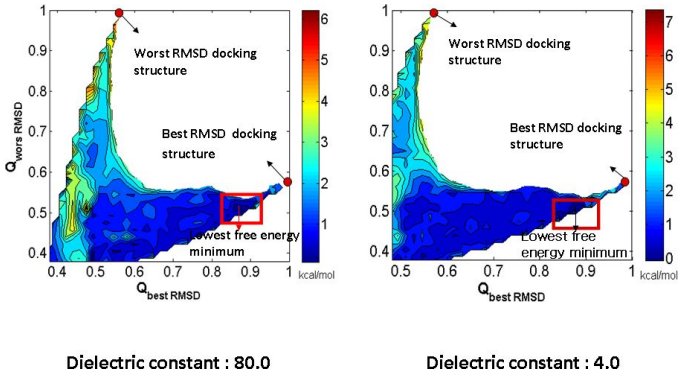


Fig. 2. 2-dimensional free energy surface of ColE7-Im7 complex for dielectric constant ϵ=80.0 and ϵ=4.0. The two red circles are best RMSD docking structure and worst docking RMSD structure. The red box corresponds to the lowest free energy minimum (Q_{best},Q_{worst}:0.878, 0.504) at ϵ=80.0 and (Q_{best},Q_{worst}:0.894, 0.520) at ϵ=4.0.

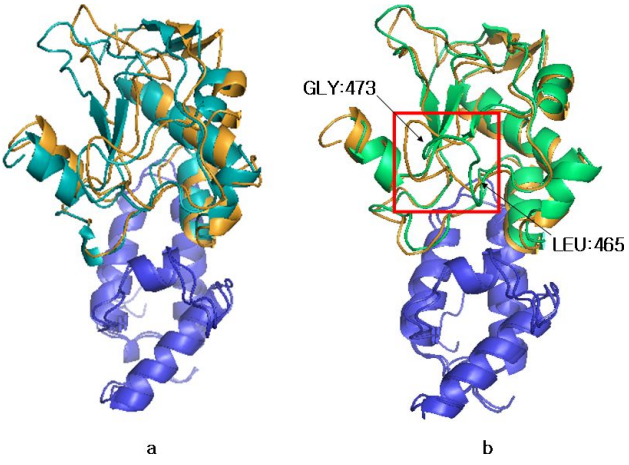


Fig. 3. Figure 3.a shows the superimposed images of best RMSD (Cyan) and the lowest free energy minimum structure (Orange) at dielectric constant ϵ=4.0. Figure 3.b shows the superimposed images of the the lowest free energy minimum structure (Orange) and the X-ray crystal (Green) dielectric constant ϵ=4.0. The Im7 (Inhibitor) is shown as blue color.

Leu465 to Gly473 in red box in Fig. 3b. The buried solvent accessible surface areas (SASA) were calculated for the X-ray crystal structure (1381 Å^2), the best RMSD docking structure (1434 Å^2), and the lowest free energy minimum structure at ϵ=4.0 (1787 Å^2)[2]. It indicates the lowest free energy minimum structure is more tightly packed at the interface of the ColE7-Im7 complex than the X-ray crystal structure.

4 Discussion

We performed free energy calculation in the 2-dimensional conformation space composed of the two references, best and the worst RMSD structures. The docking method and the free energy sampling method are quite complementary to each other. The docking method (based on the rigid body docking) dramatically narrows the sampling space. Otherwise, the blind test would require exhaustive computational performance. If the X-ray crystal structure of the protein-protein complexes is identified (7CEI in our study), we can test the validity of the docking method. By calculating the RMSD value with respect to the X-ray crystal structure, we can choose two references of the best and worst RMSD docking structures. Also, if we also have biological information (such as catalytic site or active site), physical (main groove or minor grove) and chemical information (hydrophobic, hydrophilic, H-bonds) about the protein-protein complex, the docking method remains an essential tool to predict the "nearly correct" configuration of

[2] 1.4 Å was used as the probe radius for the calculation of SASA.

the protein-protein complexes within the rigid body approximation. However, if the X-ray crystal form of the protein-protein complex is not available, or when the sufficient information about the chemical and physical composition of each entity in protein-protein complexes is missing, the rigid docking method would be less reliable. Furthermore, when flexible loops play critical roles in forming protein-protein complexes, the error in the rigid body prediction will increase. In this case, however, the docking method provide several "*plausible*" candidates for the protein-protein complex at the rough prediction level. The free energy method then performs the key mission of finding the "nearly correct structure" from the "plausible" structures, including information of flexible loops. As a future study, we will apply our method with explicit water to the present case and finally to the case for which an X-ray crystal of the complex has not been solved.

Acknowledgements

LGP acknowledges support from NIH-06350, NSF FRG DMR 0804549, the intramural research program of the NIH and National Institute of Environmental Health Sciences.

References

1. Smith, G.R., Stenberg, M.J.E.: Prediction of Protein-Protein Intreactions by Docking Methods. Curr. Opin. Struct. Biol. 12, 28–35 (2002)
2. Ritchie, D.W.: Recent Progress and Future Directions in Protein-Protein Docking. Curr. Prot. Pepti. Sci. 9, 1–15 (2008)
3. Katchalski-Katzir, E., Sharive, I., Eisenstein, M., Friesem, A.A., Aflalo, C., Vakser, I.A.: Molecular Surface Recognition: Determination of Geometric Fit between Proteins and Their Ligands by Correlation Technique. Proc. Natl. Acad. Sci. USA. 89, 2195–2199 (1992)
4. Gabb, H.A., Jackson, R.M., Sternberg, M.J.E.: Modelling Protein Docking Using Shape Complementarity, Electrostatics and Biochemical Information. J. Mol. Biol. 272, 106–120 (1997)
5. Zhang, C., Vasmatzis, G., Cornette, J.L., DeLisi, C.: Determination of Atomic Desolvation Energies from the Structures of Crystallized Proteins. J. Mol. Biol. 267, 707–726 (1997)
6. Chen, R., Weng, Z.: Docking Unbound Proteins Using Shape Complementarity, Desolvation, and Electrostatics. Proteins 47, 281–294 (2002)
7. Berchanski, A., Shapira, B., Eisenstein, M.: Hydrophobic Complementarity in Protein-Protein Docking. Proteins 56, 130–142 (2004)
8. Kozakov, D., Brenke, R., Comeau, S.R., Vajda, S.: PIPER: an FFT-Based Protein Docking Program with Pairwise Potentials. Proteins 65, 392–406 (2006)
9. Claußen, H., Buning, C., Rarey, M., Lengaue, T.: FlexE:Efficeient Molecular Docking Considering Protein Structure Variations. J. Mol. Biol. 308, 377–395 (2001)
10. Ewing, T.J.A., Makino, S., Skillman, A.G., Kuntz, I.D.: DOCK 4.0:Search Strategies for Automated Molecular Docking of Flexible Molecular Databases. J. Comput. Aided. Mol. Des. 15, 411–428 (2001)

11. Smith, G.R., Fitzjohn, P.W., Page, C.S., Bates, P.A.: Incorporation of Flexibility into Rigid-Body Docking: Applications in Rounds 3-5 of CAPRI. Proteins 60, 263–268 (2005)
12. Zacharias, M.: Rapid Protein-Ligand Docking Using Soft Modes from Molecular Dynamics Simulations to Account for Protein Deformability:Binding of FK506 to FKBP. Proteins 54, 759–767 (2004)
13. Roux, B.: The Calculation of the Potential of Mean Force Using Computer Simulation. Comput. Phys. Comm. 91, 275–282 (1994)
14. Kumar, S., Rosenberg, J.M., Bouzida, D., Swendsen, R.H., Kollman, P.A.: The Weighted Histogram Analysis Method for Free-Energy Calculation on Biomolecules. I. The Method. J. Comput. Chem. 13, 1011–1021 (1992)
15. Ferrenberg, A.M., Swendsen, R.H.: Optimized Monte Carlo Data Analysis. Phys. Rev. Lett. 63, 1195–1198 (1989)
16. Banavali, N.K., Roux, B.: Free Energy Landscape of A-DNA to B-DNA Conversion in Aqueous Solution. J. Am. Chem. Soc. 127, 6866–6876 (2005)
17. Arora, K., Brooks III, C.L.: Large-Scale Conformational Transitions of Adenylate Kinase Appear to Involve a Population-Shift Mechanism. Proc. Natl. Acad. Sci. USA 104, 18496–18501 (2007)
18. Pugsley, A.P., Oudega, B.: Methods for Studing Colicins and Their Plasmids. In: Hardy, K.G. (ed.) Plasmids: A Practical Approach, pp. 105–161. IRL Press, Oxford (1987)
19. Ko, T.P., Liao, C.C., Ku, W.Y., Chak, K.F., Yuan, H.S.: The Crystal Structure of the DNAase Domain of Colicin E7 in Complex with its Inhibitor Im7 Protein. Struct. 7, 91–102 (1999)
20. Wallis, R., Leung, K.Y., Pommer, A.J., Videler, H., Moor, G.R., James, R., Kleanthous, C.: Protein-Protein Interactions in Colicin E9 DNAase-Immunity Protein Complexes.2. Cognate and Noncognate Interactions That Span the Millimolar to Femtomolar Affinity Range. Biochem. 34, 13751–13759 (1995)
21. Mintseris, J., Wiehe, K., Pierce, B., Anderson, R., Chen, R., Janin, J., Weng, Z.: Protein-Protein Docking Benchmark 2.0: An Update. Proteins 60, 214–216 (2005)
22. Wolynes, P.G.: Landscapes, Funnesls, Glasses, and Folding: from Metaphors to Software. Proc. Am. Phil. Soc. 145, 555–563 (2001)
23. Takagi, F., Koga, N., Takada, S.: How Protein Thermodynamics and Folding Mechanisms are Altered by the Chaperonin Cage:Molecular Simulations. Proc. Natl. Acad. Sci. USA 100, 11367–11372 (2003)
24. Wu, S., Zhuravlev, P.I., Papoian, G.A.: High Resolution Approach to the Native State Enesemble Kinetics and Thermodynamics. Biophys. J. 95, 5524–5532 (2008)
25. Plimton, S.J.: Fast Parallel Algorithms for Short-Range Molecular Dynamics. J. Comput. Phys. 117, 1–19 (1995)

The Design of Tris(*o*-phenylenedioxy)cyclotrisphosphazene (TPP) Derivatives and Analogs toward Multifunctional Zeolite Use

Godefroid Gahungu, Wenliang Li, and Jingping Zhang

Faculty of Chemistry, Northeast Normal University, Changchun 130024, China
zhangjingping66@yahoo.com.cn

Abstract. Taking tris(*o*-phenylenedioxy)cyclotrisphosphazene (TPP) as template, series of derivatives and analogs were designed with the aim to investigate the structural features of organic zeolite (OZ) and their potential applications. On the basis of DFT-PBE0/6-31G** quantum calculation, the results show a tight dependence of the electron donor (E-D) of the entire molecule on that of the side group and bridge. It was found that extending the side fragment with a phenyl ring and substituting CH/N, or tetrathiafulvalene (TTF)-like group, or the side phenyl fragments substitution by TTF and its derivatives, preserve the "paddle wheel" molecular shape, a key factor in the tunnel formation on which is based the organic OZ use of TPP. In comparison with the commonly used organic superconductors, most of the designed molecules with TTF fragments were predicted to show comparable or better E-D strength.

Keywords: molecular design, DFT, organic zeolite, TPP, TTF.

1 Introduction

The absorption properties of materials are emerging as a forefront issue of present day research, due to the strategic industrial and environmental applications, such as gas storage, selective gas recognition, and separation [1]. As a result of their unique features, molecular self-assembled materials and organic zeolites (OZ) [2, 3] seem to constitute a competing alternative in this field, and are thus still to be explored extensively. Originally studied by Allcock [4], tris(*o*-phenylenedioxy)cyclotriphosphazene (TPP, 1a) became a compound of choice to investigate the structural features of OZ and their potential applications.

Studies focused on the stability of the hexagonal modification compared to compact guest-free monoclinic [5], the investigation of gas storage or aromatic guest insertion by advanced NMR techniques [6], the confinement of I_2 molecules by several crystallization procedures [7], and the insertion of dipolar molecules [8]. From TPP to some of its derivatives, it has been shown that the available space for absorbates can be modulated by the choice of the side group, which substitutes the dioxyphenylene in the former, the key-factor of the tunnel formation being reported to be the rigid "paddle wheel" molecular shape and the requirements of the crystal state [9]. For example, it was reported that TPP and tris(2,3-dioxynaphthyl)cyclotri-phosphazene, (TNP, 2a)

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 229–237, 2009.

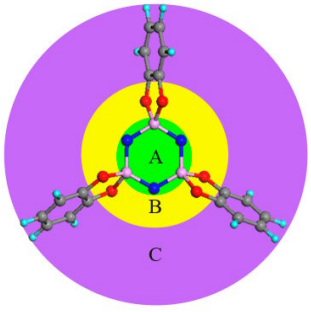
spontaneously form inclusion adducts with benzene, toluene, heptane, octanes, and many other compounds [4, 10, 11]. With 1a and some of its derivatives, clathration was characterized as a pure mechanical phenomenon [12]. Some of its relevant applications however, may be based on physico-chemical properties. Within TPP zeolite, which shows a strong affinity to include gaseous CH_4, CO_2 [6b], I_2 and Xe [6a, 7, 13], specific host-guest interactions of the donor-acceptor type are expected for channels. Recent report by Hertzsch T. has shown that 1a may be used to remove radioactive I_2 [6b], even from a humid environment or water [14]. The stability of the inclusion compound $TPP(I_2)_{0.75}$ up to 420 K [7], was interpreted based on the Lewis acidity of I_2 and the electron-donor (E-D) capacity of the TPP-phenylenedioxy rings. It appears clear that the E-D capacity may play a certain role in the trapping process of some compounds within TPP OZ, which may provide some potential applications in the environmental chemistry. From this viewpoint, different TPP-like materials (Fig. 1) are studied in this contribution. We focused our interest on the relationship between the E-D capacity of the side fragment (part C), bridge (part B), and that for the entire molecule. Although a number of theoretical works on phosphazene containing systems can be found in the literature [15, 16], very few were devoted to related OZ and the relationship between E-D and the structure of molecules [16]. This contribution may provide some helpful insights toward the understanding of TPP-like OZ uses and the further design as well.

2 Computational Strategy

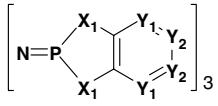
Taking TPP or TNP molecular structure as template, we have designed series of TPP or TNP analogs by systematically extending or substituting the side fragment with a phenyl ring and substituting CH/N, or tetrathiafulvalene (TTF)-like group, or the bridge O substituted by NH totally or partially as described in Fig. 1. All molecular geometry optimizations were carried out with the aid of Gaussian 03 package [17]. During the geometry optimization, the neutral species were constrained within the C_3 symmetry. Density functional theory (DFT) calculation using the PBE0 functional [18] with the 6-31G(d,p) [19] basis set was performed in the geometry optimization, which was proved to be proper for such kind of system [16a]. The equilibrium structures were located using analytical energy derivatives. To confirm the structure is a minimum on the potential energy surface at this level of theory, frequency calculations were performed. The unrestricted formalism was used for the oxidized forms and uniformly estimated from PBE0/6-31+G (d, p) calculation including diffuse functions needed to describe the cations. From the calculated $<S^2>$ values, the spin contamination included in the present calculation results was confirmed to be in general no more than 3%. IP of the molecules were calculated as described in the equation below:

$$IP = -E_{HOMO}. \tag{1}$$

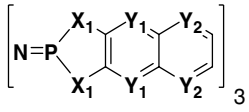
Where E_{HOMO} is the HOMO energy (according to the Koopman`s theorem [20]) at the HF/6-31G(d,p) level. Our preliminary calculations have proved that based on Koopman`s theory, the HF/6-311+G* can yield an excellent accuracy for adiabatic IP [16].



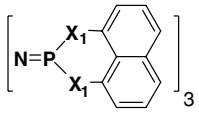
TPP



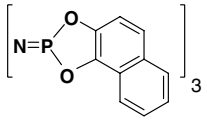
1



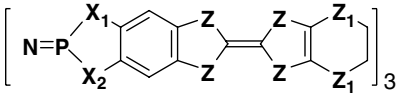
2



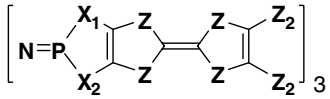
3



4



5



6

a: (X_1, X_2) = (O, O); **b**: (X_1, X_2) = (O, NH); **c**: (X_1, X_2) = (NH, NH);
1a, **2a**: **d**: (Y_1, Y_2) = (CH, N); **e**: (Y_1, Y_2) = (N, CH);
5, **6**: **f**: Z = S; **g**: Z= Se; **h**: Z_1 = O; **i**: Z_1 = S; **j**: Z_2 = H; **k**: Z_2 = CH_3

Fig. 1. Chemical structures of the derivatives and analogs of TPP under investigation

3 Results and Discussion

It is found that the geometry of the core ring is independent of the variation of bridge atom and/or side fragment. All the structures of investigated compounds preserve the "paddle wheel" molecular shape (as schematically shown in Fig. 2 as examples), a key factor in the tunnel formation on which is based the organic OZ use of TPP as soft material. In general, a good agreement was found between the calculated structures and available crystal data [5, 12a]. The frontier molecular orbital (FMO) distribution of investigated compounds all localized on their three spirocyclic side groups, as shown in Fig. 3. The E-D strength of the phenyl ring within TPP affects the stability of the adsorbent···adsorbate complex [7]. Therefore, we demonstrated herein a tight relation between the IP of TPP–like molecules and that of the free side fragment. Then, one may directly estimates the effect of the substitution on the bridge; the CH/N heterosubstitution on the side fragment; the extending of side group by extra phenyl ring or TTF-like fragment; and the substitution of phenyl by TTF-like fragments for the E-D capacity of TTP and its analogs from the IP of the entire molecules. The obtained electronic properties such as the IP, FMO energies, and energy gaps

between HOMOs and LUMOs are summarized in Table 1. Furthermore, the optimized structures for the neutral and cationic species for investigated compounds only revealed small structural changes, suggesting being the good candidates for OZ use.

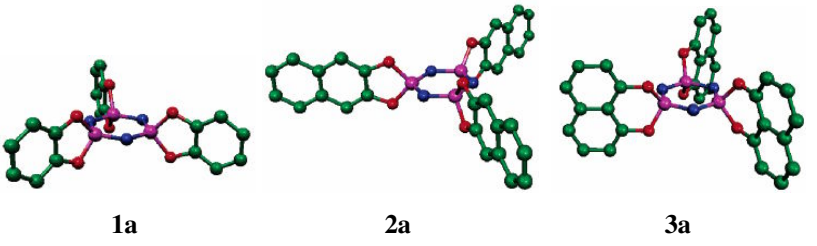


Fig. 2. Optimized geometries for 1a-3a (for clarity, hydrogen atoms are not shown)

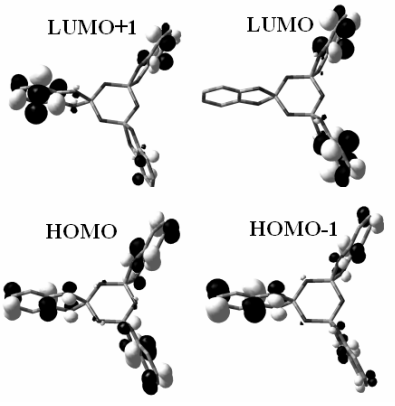


Fig. 3. Frontier Molecular Orbitals for 1a computed at PBE0/6-31G(d,p)

3.1 The Effect of Enhanced π Conjugation and the Bridge O/NH Partially or Totally Substitution

We begin our discussion with the already synthesized compounds (1a-3a) and 4a which correspond to extending the phenylenedioxyl side group (within TPP) with one more phenyl rings, linearly (2a) or laterally (3a, 4a). Some of the corresponding optimized structures are displayed in Fig. 2. In agreement with experimental observations is the planarity of the side fragment in the cases of 1a, 2a, and of course, the twisted heterocycle (containing the two O atoms) in 3a. With the aim of evaluating the influence of CH/N heterosubstitution on the molecular structure of TPP and TPP-like molecules, compounds 1a and 2a were considered for this issue.

As clearly summarized in Fig. 4, the results suggest that (i) the O/NH substitution increases both the E_{HOMO} and E_{LUMO} energies in the sequence Na < Nb < Nc (with N = 1, 2, and 3) and (ii) π-conjugation increases the E_{HOMO}, while decreasing the E_{LUMO} in the sequence of 1i < 2i < 3i (i = a, b, and c). From these results, it may be concluded

that comparatively to 1a, extending the side group with an aromatic ring destabilizes the HOMO, which becomes more stabilized by the O/NH substitution. A comparison of 2i to 3i (or 4a) shows that the HOMO is even more destabilized by a lateral extension. E-D capacity is then increased within the same order as confirmed by IP calculations whose results are summarized in Table 1. The results show a tight dependence of the E-D capacity of the TPP-like molecules on that of the free side group, resulting in some interesting implications for some aspects of OZ use: (i) The stability of the inclusion compound, $TPP(I_2)_x$, and the operating temperatures may be improved by using 1c, 2a, and 3a whose clathrates with many other molecules are already known, with the OZ-I_2 inclusion compound based on 2a being expected to be less stable than that based on 3a. (ii) The E-D capacity of TPP side groups appears to be tunable, allowing predictions to be made about the stability of the inclusion compounds of OZ and molecules of Lewis acidity comparable to that of I_2.

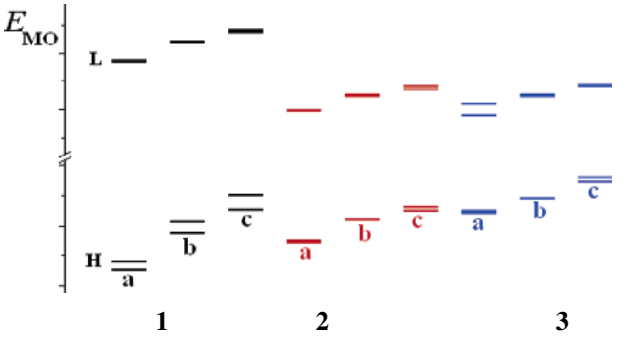


Fig. 4. PBE0 frontier molecular orbital (FMO) energy diagram for 1a-c, 2a-c, and 3a-c (L = LUMOs, H = HOMOs)

3.2 The CH/N Heterosubstitution Effect on the Side Fragment

With the aim to evaluate the influence of the CH/N hetero-substitution on the molecular structure of TPP and TPP-like molecules, compounds 1a, 2a and their CH/N substituted derivatives (1ad-e, 2ad-e) were considered for this issue. From the length of the side fragment viewpoint, on which depends the available space for adsorbates, we anticipate a decreasing diameter of the tunnel with the degree of CH/N substitution within both TPP and TNP.

This can be explained by comparing the C-C bond length of the unsubstituted bond (1.37-1.41 Å) to that of the corresponding C-N one (1.28-1.30 Å) in the CH/N-substituted derivatives. On the basis of our results the magnitude of the variation is expected in the decreasing order unsubsituted > disustituted derivatives. Thus, the size of the adsorbing space in CH/N derivatives of TPP may be anticipated to be smaller than that of TPP itself, while CH/N derivatives of TNP would lead to crystals having a tunnel spacein between those with TPP and TNP.

Table 1. The obtained electronic properties: the IP_{KT}, FMO energies, and Eg between HOMO and LUMO (eV)

compounds	IP_{KT}	E_{HOMO}	E_{LUMO}	Eg
1a	8.79	-6.60	-0.15	6.45
1b	8.02	-5.93	0.18	6.11
1c	7.53	-5.49	0.37	5.86
2a	8.00	-6.25	-1.04	5.21
2b	7.66	-5.90	-0.78	4.12
2c	7.46	-5.69	-0.64	5.05
3a	7.68	-5.76	-1.12	4.64
3b	7.26	-5.55	-0.78	4.77
3c	6.91	-5.20	-0.58	4.62
4	7.92	-6.69	0.01	6.69
1ad	10.94	-8.77	-0.35	8.42
1ae	9.80	-8.40	-0.48	7.92
2ad	9.11	-7.91	-0.90	7.01
2ae	9.21	-7.98	-0.86	7.12
5afh	7.02	-4.91	-1.03	3.88
5afi	7.13	-5.11	-1.01	4.10
5agi	7.16	-5.19	-1.38	3.80
5bfh	6.85	-4.77	-0.90	3.87
5bfi	6.93	-4.96	-0.89	4.08
5bgi	6.99	-5.05	-1.26	3.79
5cfh	6.72	-4.76	-0.83	3.86
5cfi	6.82	-4.87	-0.81	4.06
5cgi	6.85	-4.94	-1.20	3.75
5afh	7.02	-4.91	-1.03	3.88
5afi	7.13	-5.11	-1.01	4.10
6afj	7.18	-5.08	-1.14	3.94
6afk	7.04	-4.92	-1.02	3.90
6agj	7.13	-5.13	-1.50	3.63
6agk	7.00	-4.98	-1.40	3.58
6bfj	7.02	-4.95	-0.95	4.00
6bfk	6.89	-4.79	-0.84	3.95
6bgj	6.98	-4.99	-1.33	3.66
6bgk	6.87	-4.86	-1.23	3.63
6cfj	6.81	-4.74	-0.92	3.82
6cfk	6.69	-4.61	-0.81	3.80
6cgj	6.80	-4.81	-1.29	3.53
6cgk	6.69	-4.68	-1.20	3.48

The CH/N substitution in the side fragment decreases (and stabilizes) the HOMO and LUMO eigenvalues at the same time owing to the presence of two nitrogen atoms and very dependently on the position of the substituted CH group in the side fragment. Due to the inductive effect of the nitrogen atom, the HOMO gets stabilized in the sequence 1a < 1ae < 1ad within the subgroup of TPP and its CH/N derivatives and in the sequence of 2a < 2ad < 2ae in the subclass of TNP and its CH/N derivatives. The predicted net effect was that, in comparison to TPP, extending the side group with an aromatic ring (TNP) destabilizes the HOMO, which becomes more stabilized by CH/N substitution.

3.3 The Effects of TTF-Like Fragments in the Side Groups

To design novel materials combining a good E-D strength and "paddle wheel" molecular shape responsible for inclusion adducts formation, we introduce TTF-like fragments in to TPP by fused with phenyl ring (5) or substituting it (6), which may lead to potential candidates for superconductors, that may combine a good electron-donor ability and a possible inclusion adduct formation. The bridging parts by O/NH substitution are also considered.

In general, most of the new derivatives are predicted to preserve the "paddle wheel" molecular shape, the TTF-containing side group retaining the TTF-like donor behavior (TTF-like moiety distortion into the planar) during the oxidization process. From the electron-donor ability point of view, the current study shows clearly that, comparatively to the commonly used electron donors, such as TTF, Tetramethyltetraselenafulvalene (TMTSF), Bis(ethylenedithio)tetrathiafulvalene (ET), Bis(ethylenedioxy)tetrathiafulvalene (BETS-TTF), and Bisethylenedioxytetrathiafulvalene) (BO), whose predicted IP_{KT} ranges from 6.65-7.05 eV, a comparable E-D ability can be reached by adopting the building approach developed in this work (whose predicted IP_{KT} ranges from 6.69-7.18 eV). An additional insight provided by the current results is that the O/NH substitution induces a relatively significant decrease in the IP, increasing therefore the E-D strength. Finally, one may find from the same results that partial substitution leads to TTF-containing TPP analogs whose IP values (E-D strength) may be in between those of the corresponding derivatives from the total O/NH.

4 Conclusion

Using TPP as template, series derivatives or analogs were designed by chemical modification of TPP by the substitution of bridge part or side fragments. On the basis of DFT-BPE0/6-31G** quantum calculation, the results show a tight dependence of the E-D of the entire molecule on that of the side groups and bridge part, which may result in some interesting implications for some aspects of OZ use, i.e., (i) The stability of the inclusion compound, OZ-I_2, and the operating temperatures may be improved by using high E-D material (ii) The E-D capacity of TPP side groups appears to be tunable, allowing predictions to be made about the stability of the inclusion compounds of OZ and molecules of Lewis acidity comparable to that of I_2. It was concluded that the total O/NH substitution for the bridge part may significantly enhance the electron-donor capacity without altering the tolerance of TPP-like host materials to the guest molecules. The E-D capacity was found to be more significantly enhanced by a lateral than a linear extension with phenyl ring, while it decreased upon CH/N heterosubstitution, which can affect the stability of some related host·····guest complexes in the same order. The extension (or substitution) of the phenylenedioxyl group with an aromatic ring especially by introducing TTF fragments significantly enhance the E-D. In addition, in comparison with the commonly used organic superconductors, most of the designed molecules with TTF fragments were predicted to show comparable or better E-D strength, suggesting them to be good candidates for organic superconductors.

Acknowledgments. Financial supports from the NSFC (Nos. 50873032, 20773022), the NCET-06-0321, the JLSDP (20082212), and the NENU-STB-07-007 are gratefully acknowledged.

References

1. Chae, H.K., Siberio-Perez, D.Y., Kim, J.Y., Eddaoudi, G.M., Matzger, A.J., Keeffe, M.O., Yaghi, O.M.: A Route To High Surface Area, Porosity and Inclusion of Large Molecules In Crystals. Nature 427, 523–527 (2004); Ward, M.D.: Enhanced: Molecular Fuel Tanks. Science 300, 1104–1105 (2003); Kuznicki, S.M., Bell, V.A., Nair, S., Hillhouse, H.W., Jacubinas, R.M., Braunbarth, C.M., Toby, B. H., Tsapatsis, M.: A Titanosilicate Molecular Sieve with Adjustable Pores for Size-Selective Adsorption of Molecules. Nature 412, 720–724 (2001)
2. Blau, W.J., Fleming, A.J.: Designer Nanotubes by Molecular Self-Assembly. Science 304, 1457–1458 (2004)
3. Whitesides, G., Grzybowski, M.B.: Self-Assembly at All Scales. Science 295, 2418–2421 (2002)
4. Allcock, H.R., Siegel, L.A.: Phosphonitrilic Compounds. III. Molecular Inclusion Compounds of Tris(*o*-phenylenedioxy)phosphonitrile Trimer. J. Am. Chem. Soc. 86, 5140–5144 (1964)
5. Allcock, H.R., Levin, M.L., Whittle, R.R.: Tris(*o*-phenylenedioxy)cyclotriphosphazene: The Clathration-Induced Monoclinic to Hexagonal Solid State Transition. Inorg. Chem. 25, 41–47 (1986)
6. Sozzani, P., Comotti, A., Simonutti, R., Meersmann, T., Logan, J.W., Pines, A.: A Porous Crystalline Molecular Solid Explored by Hyperpolarized Xenon. Angew. Chem. Int. Ed. 39, 2695–2699 (2000); Sozzani, P., Bracco, S., Comotti, A., Ferretti, L., Simonutti, R.: Methane and Carbon Dioxide Storage in a Porous van der Waals Crystal. Angew. Chem. Int. Ed. 44, 1816–1820 (2005); Sozzani, P., Comotti, A., Bracco, S., Simonutti, R.: A Family of Supramolecular Frameworks of Polyconjugated Molecules Hosted in Aromatic Nanochannels. Angew. Chem., Int. Ed. 43, 2792–2797 (2004)
7. Hertzsch, T., Budde, F., Weber, E., Hulliger, J.: Supramolecular-Wire Confinement of I_2 Molecules in Channels of the Organic Zeolite Tris(*o*-phenylenedioxy)cyclotriphosphazene. Angew. Chem. Int. Ed. 41, 2281–2284 (2002)
8. Hertzsch, T., Kluge, S., Weber, E., Budde, F., Hulliger, J.: Surface Recognition of Dipolar Molecules Entering Channels of the Organic Zeolite Tris(*o*-phenylenedioxy)cyclotriphosphazene. Adv. Mater. 13, 1864–1867 (2001)
9. Allcock, H.R., Stein, M.T., Stanko, J.A.: The Crystal and Molecular Structure of Tris(2,2'-dioxybiphenyl)cyclotriphosphazene. J. Am. Chem. Soc. 93, 3173–3178 (1971)
10. Allcock, H.R., Stein, M.T.: Clathration by Tris (2,3-naphthalenedioxy)cyclotriphosphazene. An X-ray Crystal and Molecular Structure Study. J. Am. Chem. Soc. 96, 49–52 (1974)
11. Allcock, H.R., Kugel, R.L.: Cyclized Products From the Reactions of Hexachlorocyclotriphosphazene (Phosphonitrilic Chloride Trimer) with Aromatic Dihydroxy, Dithiol, and Diamino Compounds. Inorg. Chem. 5, 1016–1020 (1966)
12. Siegel, L.A., Van den Hende, J.H.: The Crystal Structure of Molecular Inclusion Compounds of tris(*o*-phenylenedioxy)phosphonitrile Trimer. J. Chem. Soc. A., 817–820 (1967); Allcock, H.R., Allen, R.W., Bissel, E.C., Smeltz, L.A., Teeter, M.: Molecular Motion and Molecular Separations in Cyclophosphazene Clathrates. J. Am. Chem. Soc. 98, 5120–5125 (1976)

13. Couderc, G., Hertzsch, T., Behrnd, N.-R., Kramer, K., Hulliger, J.: Reversible Sorption of Nitrogen and Xenon Gas by the Guest-free Zeolite Tris(*o*-phenylenedioxy)cyclotriphosphazene (TPP). Microporous and Mesoporous Materials 88, 170–175 (2006)
14. Hertzsch, T., Gervais, C., Hulliger, J., Jaeckel, B., Guentay, S., Bruchertseifer, H., Neels, A.: Open-Pore Organic Material for Retaining Radioactive I_2 and CH_3I. Adv. Funct. Mater. 16, 268–272 (2006)
15. Breza, M.: On bonding in Cyclic Triphosphazenes. J. Mol. Struct (Theochem.) 505, 169–177 (2000); Breza, M.: The Electronic Structure of Planar Phosphazene Rings Polyhedron 19, 389–397 (2000); Luana, L., Pendas, A.M., Costales, A.: Topological Analysis of Chemical Bonding in Cyclophosphazenes. J. Phys. Chem. A. 105, 5280 (2001); Waltman, R.J., Lengsfield, B., Pacansky, J.: Lubricants for Rigid Magnetic Media Based upon Cyclotriphosphazenes : Interactions with Lewis Acid Sites. Chem. Mater. 9, 2185–2196 (1997); Gahungu, G., Zhang, B., Zhang, J.: Theoretical Study of Tris(o-phenylenedioxy) cyclotrisphosphazene (TPP) Electronic Structure with Ab Initio and DFT methods. Chem. Phys. Lett. 388, 422–426 (2004); Gervais, C., Hertzsch, T., Hulliger, J.: Insertion of Dipolar Molecules in Channels of a Centrosymmetric Organic Zeolite: Molecular Modeling and Experimental Investigations on Diffusion and Polarity Formation. J. Phys. Chem. B. 109, 7961–7968 (2005)
16. Gahungu, G., Zhang, B., Zhang, J.P.: Influence of the Substituted Side Group on the Molecular Structure and Electronic Properties of TPP and Related Implications on Organic Zeolites Use. J. Phys. Chem. B 111, 5031–5033 (2007); Gahungu, G., Zhang, B., Zhang, J.P.: Design of Tetrathiafulvalene-Based Phosphazenes Combining a Good Electron-Donor Capacity and Possible Inclusion Adduct Formation (Part II). J. Phys. Chem. C 111, 4838–4846 (2007) ; Gahungu, G., Zhang, J. P.: Design of TTF-Based Phosphazenes Combining a Good Electron-Donor Capacity and Possible Inclusion Adduct Formation. J. Phys. Chem. B 110, 16852–16859 (2006)
17. Frisch, M.J., Trucks, G.W., Schlegel, H.B., Scuseria, G.E., Robb, M.A., Cheeseman, J.R., Montgomery Jr., J.A., Vreven, T., Kudin, K.N., Burant, J.C., Millam, J.M., Iyengar, S.S., Tomasi, J., Barone, V., Mennucci, B., Cossi, M., Scalmani, G., Rega, N., Petersson, G.A., Nakatsuji, H., Hada, M., Ehara, M., Toyota, K., Fukuda, R., Hasegawa, J., Ishida, M., Nakajima, T., Honda, Y., Kitao, O., Nakai, H., Klene, M., Li, X., Knox, J.E., Hratchian, H.P., Cross, J.B., Bakken, V., Adamo, C., Jaramillo, J., Gomperts, R., Stratmann, R.E., Yazyev, O., Austin, A.J., Cammi, R., Pomelli, C., Ochterski, J.W., Ayala, P.Y., Morokuma, K., Voth, G.A., Salvador, P., Dannenberg, J.J., Zakrzewski, V.G., Dapprich, S., Daniels, A.D., Strain, M.C., Farkas, O., Malick, D.K., Rabuck, A.D., Raghavachari, K., Foresman, J.B., Ortiz, J.V., Cui, Q., Baboul, A.G., Clifford, S., Cioslowski, J., Stefanov, B.B., Liu, G., Liashenko, A., Piskorz, P., Komaromi, I., Martin, R.L., Fox, D.J., Keith, T., Al-Laham, M.A., Peng, C.Y., Nanayakkara, A., Challacombe, M., Gill, P.M.W., Johnson, B., Chen, W., Wong, M.W., Gonzalez, C., Pople, J.A.: Gaussian 2003, Revision B.05. Gaussian, Inc., Pittsburgh (2003)
18. Adamo, C., Barone, V.: Toward Reliable Density Functional Methods without Adjustable Parameters: The PBE0 Model. J. Chem. Phys. 110, 6158–6170 (1999); Van Voorhis, T. Scuseria, G. E.: A Novel Form for the Exchange-correlation Energy Functional. J. Chem. Phys. 109, 400–410 (1998)
19. Hariharan, P.C., Pople, J.A.: Accuracy of AH_n Equilibrium Geometries by Single Determinant Molecular Orbital Theory. Mol. Phys. 27, 209–214 (1974); Gordon, M.S.: The Isomers of Silacyclopropane. Chem. Phys. Lett. 76, 163–168 (1980); Frisch, M. J., Pople, J.A., Binkley, J.S.: Self-consistent Molecular Orbital Methods 25. Supplementary Functions for Gaussian Basis Sets. J. Chem. Phys. 80, 3265–3269 (1984)
20. Koopmans, T.: Über die Zuordnung von Wellenfunktionen und Eigenwerten zu den Einzelnen Elektronen Eines Atoms. Physica 1, 104–113 (1934)

Atmospheric and Oceanic Computational Science
First International Workshop

Adrian Sandu[1], Amik St-Cyr[2], and Katherine J. Evans[3]

[1] Virginia Polytechnic Institute and State University
sandu@cs.vt.edu
[2] National Center for Atmospheric Research
amik@ucar.edu
[3] Oak Ridge National Laboratory
evanskj@ornl.gov

1 The Workshop

The first workshop on Atmospheric and Oceanic Computational Science brings together computational and domain scientists who develop computational tools for the study of the atmosphere and oceans. These tools are essential for understanding and prediciting weather, air and water pollution, and the evolution of the planet's climate. The dynamics of the atmosphere and of the oceans is driven by a multitude of physical processes and is characterized by a multiple spatial and temporal scales. Moreover, the computations are very large scale: present day models track the time evolution of tens of millions to tens of billions variables. These factors make atmospheric and oceanic simulations a challenging, vibrant research field with a tremendous impact on society at large.

Topics covered in this symposium include new methods for spatial and temporal discretization, parallel and high performance computing, advances with existing models, and data assimilation and observation targeting algorithms.

2 The Papers

Ten papers have been selected for oral presentations.

A Fully Implicit Jacobian-Free High-Order Discontinuous Galerkin Mesoscale Flow Solver by A. St-Cyr, D. Neckels proposes a discretization of the compressible Euler equations using the discontinuous Galerkin approach on collocated Gauss type grids. Time discretization uses a stiffly stable Rosenbrock W-method is combined with an approximate evaluation of the Jacobian.

Time acceleration methods for convection on the cubed sphere by R. Archibald, K. Evans, J. Drake, and J. White discusses new algorithms to overcome the scalability barriers in climate simulations. A combination of multiwavelet discontinuous Galerkin method with exact linear part time-evolution schemes can overcome the time barrier for advection equations on a sphere.

Comparison of Traditional and Novel Discretization Methods for Advection Models in Numerical Weather Prediction by C. Mavriplis, S. Crowell, D. Williams, and

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 241–242, 2009.

L. Wicker compares CPU time, number of degrees of freedom and overall behavior of solutions for finite difference, spectral difference and discontinuous Galerkin methods on several model advection problems relevant to numerical weather prediction.

A non-oscillatory advection operator for the compatible spectral element method by M.A. Taylor, A. St-Cyr,and A. Fournier presents the development of a monotone or sign-preserving advection operator based on a spectral element formulation, and implemented via the highly scalable cubed-sphere atmospheric dynamical core package HOMME into the Community Climate System Model (CCSM).

Simulating Particulate Organic Advection Along Bottom Slopes to Improve Simulation of Estuarine Hypoxia and Anoxia by P. Wang and L.C. Linker presents an approach to move volatile solids from the shoals to the channel by simulating movement of particulate organics due to slopes based on an example in the Chesapeake Bay eutrophication model. Implementations for the simulation of this behavior in computer parallel processing are discussed.

Explicit time stepping methods with high stage order and monotonicity properties by E.M. Constantinescu and A. Sandu introduces a three and a four order explicit time stepping methods with high stage order and favorable monotonicity properties. The proposed methods are based on general linear methods, and are generalizations of both Runge-Kutta and linear multistep methods.

Improving GEOS-Chem Model Tropospheric Ozone through Assimilation of Pseudo Tropospheric Emission Spectrometer Profile Retrievals by K. Singh, P. Eller, A. Sandu, K. Bowman, D. Jones, and M. Lee discusses a recently-developed adjoint model of GEOS-Chem global chemical transport model, and 4D-variational data assimilation studies used to improve of 2006 summer time distribution of global tropospheric ozone through assimilation of pseudo profile retrievals from the Tropospheric Emission Spectrometer (TES).

Chemical Data Assimilation with CMAQ: Continuous vs. Discrete Advection Adjoints by T.Y. Gou, K. Singh, and A. Sandu discusses a new implementation of the adjoint of Community Multiscale Air Quality (CMAQ) modeling system. The construction of discrete adjoint code is derived from the forward model code with the aid of automatic differentiation.

A Second Order Adjoint Method to Targeted Observations by H.C. Godinez and D.N. Daescu studies the role of the second order adjoints in observation targeting strategies. The dominant eigenvectors of the Hessian matrix indicate the directions of maximal error growth for a given targeting functional. These vectors are a natural choice to be included in the targeting strategies given their mathematical properties.

A scalable and adaptable solution framework within components of the Community Climate System Model by K.J. Evans, D.W.I. Rouson, M.A. Taylor, A.G. Salinger, W. Weijer, and J.B. White III implements a a framework for a fully implicit solution method into the High Order Methods Modeling Environment (HOMME), and the Parallel Ocean Program (POP) model of the global ocean. Both of these models are components of the Community Climate System Model (CCSM).

A Fully Implicit Jacobian-Free High-Order Discontinuous Galerkin Mesoscale Flow Solver

Amik St-Cyr[1,⋆] and David Neckels[2]

[1] National Center for Atmospheric Research (NCAR), Boulder, USA
amik@ucar.edu
[2] Previously NCAR now Beckman Coulter Inc., Fullerton, California, USA

Abstract. In this work it is shown how to discretize the compressible Euler equations around a vertically stratified base state using the discontinuous Galerkin approach on collocated Gauss type grids. A stiffly stable Rosenbrock W-method is combined with an approximate evaluation of the Jacobian to integrate in time the resulting system of ODEs. Simulations with fully compressible equations for a rising thermal bubble are performed. Also included are simulations of an inertia gravity wave in a periodic channel. The proposed time-stepping method accelerates the simulation times with respect to explicit Runge-Kutta time stepping procedures having the same number of stages.

1 Introduction

With modern climate models currently able to reach nonhydrostatic resolutions, it is generally perceived that the next generation of general circulation models will be able to run globally for both weather and climate prediction. The Reynolds numbers involved in global and mesoscale atmospheric modeling explains the choice of numerical weather prediction (NWP) centers for the compressible Euler equations. The eventual availability of petascale calculators forces the research stream into highly-scalable numerical methods. A very popular approach nowadays for applications is the so called discontinuous Galerkin method which enjoys some of the properties of finite-volumes and finite-elements methods [1]. It benefits from a variable polynomial degree within each element, it is highly scalable since the cost of parallel communications can be almost completely hidden [2]. For global atmospheric computations such comparable methods are performing very well [3]. However, at increasingly finer resolutions, time-stepping the compressible Euler equations is problematic. Most models utilize an explicit time-stepping procedure and are thus strongly dependent on the Courant-Friedrich-Lewy (CFL) condition which dictates the maximum allowable time-step to be proportional to the minimal spatial mesh size. The consequences of such techniques are such that, in order to solve a global climate problem at a

⋆ NCAR is operated by the University Corporation for Atmospheric Research and sponsored by the National Science Foundation (NSF). A part of this work supported by the NSF under CMG grant 0530845.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 243–252, 2009.

$1km$ resolution at scientifically relevant integration rates, a 2^{20} folds increase in computing power will be required. Following Moore's Law such a supercomputer will be available in 30 years. Therefore developments in new algorithms are of utmost importance.

In the present work a fully implicit framework for solving the compressible Euler equations at low Mach number is proposed. First, the governing equations are presented with their non-dimensionalisation. The space discretization then follows. Next, the time-discretization is discussed and the Rosenbrock W-method is introduced with the low Mach treatment of the numerical flux. Finally, various standard numerical experiments proposed to test mesoscale flow solvers are presented.

2 Governing Equations

The model system of equations that will be solved in this work is a system of m-conservation laws with source term, in two spatial dimensions, written generically as

$$\mathrm{U}_t + \nabla \cdot \mathbf{F}(\mathrm{U}) = S(\mathrm{U}). \tag{1}$$

where $\mathbf{F}(\mathrm{U}) \equiv (F, G) : \mathbb{R}^m \to \mathbb{R}^m \times \mathbb{R}^m$ and $\mathrm{U} = \mathrm{U}(\mathbf{x}, t) : \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}^m$. For convenience we define the hyperbolicity matrix as

$$A(\mathrm{U}; \hat{\mathbf{n}}) \equiv \sum_{i=1}^{2} A_i(\mathrm{U}) \hat{n}_i \tag{2}$$

with $\mathbf{A} = (A_1, A_2)$ where $A_1 = F'(\mathrm{U})$ and $A_2 = G'(\mathrm{U})$. The above generic equation permits the study of various sets of equations. In [4], the conservative form of the compressible Euler equations involves the conservation of potential temperature instead of the total energy as traditionally favored in computational fluid dynamics. The state vector and the fluxes associated with this formulation are

$$\mathrm{U} \equiv \begin{pmatrix} \rho \\ \rho u \\ \rho w \\ \rho\theta \end{pmatrix} \equiv \begin{pmatrix} \rho \\ U \\ W \\ \Theta \end{pmatrix}, \quad F \equiv \begin{pmatrix} U \\ \frac{UU}{\rho} + p \\ \frac{UW}{\rho} \\ \frac{U\Theta}{\rho} \end{pmatrix}, \quad G \equiv \begin{pmatrix} W \\ \frac{WU}{\rho} \\ \frac{WW}{\rho} + p \\ \frac{W\Theta}{\rho} \end{pmatrix}$$

and $S = (0, 0, -\rho g, 0)^T$. In the above system ρ represents density, u and w are the horizontal and vertical velocities and θ is the potential temperature. To close the system, an expression for the pressure is required:

$$p = p_{\mathrm{ref}} \left(\frac{R\Theta}{p_{\mathrm{ref}}}\right)^{\gamma} \tag{3}$$

with $\gamma = c_p/c_v = 1004.6/717.6$ and where $p_{\mathrm{ref}} = 1013.25$ kpa. For certain tests, the Eady model is required and consists into adding the $\hat{y}$ momentum equation with an f plane Coriolis force. The entire approach is still 2D and all derivatives in the $\hat{y}$ direction are zero.

The system of conservation laws is re-written around a hydrostatically balanced base state. This step is necessary in order to later avoid any kind of roundoff issues. The hydrostatic assumption reads

$$\frac{\partial \bar{p}}{\partial z} = -\bar{\rho} g.$$

Thus the following assumption is made on the dependent variables:

$$p = \bar{p}(z) + p' \tag{4}$$
$$\rho = \bar{\rho} + \rho' \tag{5}$$
$$U = \rho' u + \bar{\rho} u = U' + U \tag{6}$$
$$W = \rho' w + \bar{\rho} w = W' + W \tag{7}$$
$$\Theta = \bar{\rho}(z)\bar{\theta}(z) + \Theta'. \tag{8}$$

To minimize numerical cancellation errors associated with the original set of equations, the equation are written in their non-dimensional form. Thus a length scale x_0, a time scale t_0 along with a reference velocities u_0, pressure γp_0, ρ_0 and θ_0 are introduced into the equations using the simple replacement $\rho \to \rho\rho_0$. This leads to the apparition of the Strouhal number $\frac{x_0}{t_0 u_0}$ in front of all time derivatives and to the inverse Mach number in front of the pressure gradient. The buoyancy term has a factor $\frac{g x_0}{u_0^2}$ which can be made equal to one if $u_0 = \sqrt{g x_0}$. If a reference length x_0 is chosen, the time scale is determined by setting the Strouhal number to one. Therefore, the final form of the conserved variables and fluxes used in the discretization are

$$\mathrm{U} \equiv \begin{pmatrix} \rho' \\ (\bar{\rho}+\rho')u \\ (\bar{\rho}+\rho')w \\ (\rho\theta)' \end{pmatrix} \equiv \begin{pmatrix} \rho' \\ U \\ W \\ \Theta' \end{pmatrix}, \quad F \equiv \begin{pmatrix} U \\ \frac{UU}{(\bar{\rho}+\rho')} + \frac{1}{M^2}p' \\ \frac{UW}{(\bar{\rho}+\rho')} \\ \frac{U\Theta}{(\bar{\rho}+\rho')} \end{pmatrix}, \quad G \equiv \begin{pmatrix} W \\ \frac{WU}{(\bar{\rho}+\rho')} \\ \frac{WW}{(\bar{\rho}+\rho')} + \frac{1}{M^2}p' \\ \frac{W\Theta}{(\bar{\rho}+\rho')} \end{pmatrix}$$

The boundary conditions employed in the numerical experiments are either of the periodic or slip type. This condition is expressed simply as $\mathbf{u} \cdot \hat{\mathbf{n}} = 0$ and will be imposed weakly in the spatial discretization.

3 Spatial Discretization

The discontinuous Galerkin method can be formulated in two different ways. The first approach is one popularized by Cockburn and various authors, see [5], and is known as the *weak formulation*. The second one, the *strong formulation*, was first considered in [6] and employed herein. Also, instead of considering modal orthogonal basis functions we consider only nodal ones. Thus, in a nodal discontinuous Galerkin discretization, the computational domain Ω is partitioned into J quadrilateral elements Ω_j in which the dependent and independent variables are approximated by N–th order tensor-product polynomial expansions.

A function $U : \mathbb{R}^2 \times \mathbb{R}^+ \rightarrow \mathbb{R}$ is spatially expanded, on element Ω_j, in terms of the N–th degree Lagrangian interpolants h_i as defined in [7],

$$U_h(\mathbf{x}(r_1^j, r_2^j), t)|_{\Omega_j} \equiv \sum_{k=0}^{N} \sum_{l=0}^{N} (U_{jkl})(t)\, h_k(r_1^j(\mathbf{x}))\, h_l(r_2^j(\mathbf{x})), \tag{9}$$

where $\mathbf{x} \rightarrow (r_1^j(\mathbf{x}), r_2^j(\mathbf{x}))$ is an affine transformation, T_j, from the quadrilateral element Ω_j, on the original domain, to the reference element $[-1,1] \times [-1,1]$ and the $(\mathbf{u}_j)_{kl}$ are the nodal basis coefficients defined at the Gauss-Lobatto Legendre (GLL) quadrature points $\{\hat{\xi}_i\}_{i=0}^N$. Introducing the finite dimensional space $W_h = \{v_h \in L^2(\Omega) \quad | \quad \forall \Omega_j \in \Omega, \mathbf{v}_h \circ \ T_j \in \mathbb{P}_k\}$, the discrete weak formulation is obtained by integration by parts of the differential equation (1), using the representation (9), on an element Ω_j:

$$\begin{aligned} &\int_{\Omega_j} \mathbf{v}_h \frac{d\mathrm{U}_h}{dt} dA + \int_{\Omega_j} \mathbf{v}_h \nabla \cdot \mathbf{F}(\mathrm{U}_h) dA = \\ &\quad M \int_{\Omega_j} \mathbf{v}_h \mathrm{U}_h dA + \int_{\partial\Omega_j \setminus \partial\Omega} \mathbf{v}_h (\mathbf{F}(\mathrm{U}_h) - \mathbf{F}_{num}(\mathrm{U}_h)) \cdot \hat{\mathbf{n}} d\sigma \\ &\quad + \int_{\bar{\Omega}_j \cap \partial\Omega} \mathbf{v}_h (\mathbf{F}(\mathrm{U}_h) - \mathbf{F}_{bnd}(\mathrm{U}_h)) \cdot \hat{\mathbf{n}} d\sigma \end{aligned} \tag{10}$$

where $\mathbf{v}_h$ and U_h are in $[W_h]^4$. The flux is not uniquely defined at the boundary $\partial\Omega_j$ and a numerical flux was introduced in the above expression. The Russanov flux, or local Lax-Friedrich, is

$$\mathbf{F}_{num}(\mathrm{U}_h) \cdot \hat{\mathbf{n}} = \frac{1}{2}(\mathbf{F}(\mathrm{U}_h^R) + \mathbf{F}(\mathrm{U}_h^L)) \cdot \hat{\mathbf{n}} - \frac{1}{2}\mathcal{D}(\mathrm{U}_h^R, \mathrm{U}_h^L; \hat{\mathbf{n}})(\mathrm{U}_h^R - \mathrm{U}_h^L) \tag{11}$$

at the interface $\partial\Omega_j$ with U_h^R defined as the discrete counterpart of $\mathrm{U}^+(\mathbf{x}) \equiv \lim_{\epsilon \rightarrow 0^+} \mathrm{U}((1-\epsilon)\mathbf{x} + \epsilon\hat{\mathbf{n}})$ where $\mathbf{x} \in \partial\Omega_j$ and $\hat{\mathbf{n}}$ points in the outward direction of element Ω_j. Notice that with $\epsilon \rightarrow 0^+$, U_h^L is obtained. The matrix $\mathcal{D}$, defined in more details in a forthcoming section, is a function of ($\mathrm{U}_h^{R,L}$ and of the interface points $\mathbf{x} \in \partial\Omega_j$. Notice that the numerical flux is the only function connecting the element Ω_j to its neighboring elements. The integrals are directly evaluated using Gauss-Lobatto quadratures. The latter, for surfaces, are obtained as follows:

$$(f, g)_j \equiv \sum_{k,l=0}^{N} f(\mathbf{x}^j(\xi_k, \xi_l))|_{\Omega_k} \cdot g(\mathbf{x}^j(\xi_k, \xi_l))|_{\Omega_j} \ |\mathcal{J}^j(\xi_k, \xi_l)| \rho_k \rho_l, \tag{12}$$

where $|\mathcal{J}^j(\xi_k, \xi_l)|$ is the Jacobian of the transformation $\mathbf{x}^j(\mathbf{r})$ and $\{\rho_i\}_{i=0}^N$ are the Gauss-Legendre weights associated with the quadrature points $\{\xi_i\}_{i=0}^N$. The boundary integrals are also performed using the GLL points. Once the spatial discretization is performed, the semi-discrete problem in time is

$$\frac{d\underline{\mathrm{U}}_h}{dt} = R_h(\underline{\mathrm{U}}_h) \tag{13}$$

and a method of lines can be applied to the above system of ordinary differential equations.

4 Time Discretization

A general Rosenbrock method for solving the ODE (13) with suitable initial conditions, with an error estimator $\hat{\underline{U}}_h^{n+1}$, can be written as

$$
\begin{aligned}
r_i &= \sum_{j=1}^{i} \gamma_{ij} k_j, \quad k_i = \frac{1}{\gamma_{ii}} r_i - \sum_{j=1}^{i-1} c_{ij} r_j \\
(\frac{1}{\Delta t \gamma_{ii}} - J_h) r_i &= R_h(\underline{U}_h^n + \sum_{j=1}^{i-1} a_{ij} r_j) + \sum_{j=1}^{i-1} (\frac{c_{ij}}{\Delta t}) r_j \qquad (14) \\
\underline{U}_h^{n+1} &= \underline{U}_h^n + \sum_{j=1}^{s} m_j r_j, \quad \hat{\underline{U}}_h^{n+1} = \underline{U}_h^n + \sum_{j=1}^{s} \hat{m}_j r_j
\end{aligned}
$$

where $J_h \equiv \frac{\partial R_h}{\partial u}\Big|_{u^n}$ is simply the Jacobian. When the Jacobian J_h is exact, the above Rosenbrock method has the same stability domain as the equivalent diagonally implicit Runge-Kutta (DIRK) scheme [8]. They can be L-stable and stiffly accurate provided that the number of stages exceeds the order of the method. For an explicit Runge-Kutta, having the same number of stages s, a cost function for the average number of Krylov iterations, per stage, necessary to solve the linear systems for a Rosenbrock method can be established as $\overline{iter} \leq \frac{\Delta t_i}{\Delta t_e \; accel}$ were $\Delta t_i / \Delta t_e$ is the ratio between the implicit time-step size and the CFL constrained explicit time-step and *accel* is the desired acceleration factor with respect to explicit. The latter clearly establishes that outperforming an explicit scheme is very difficult unless very large $\Delta t_i / \Delta t_e$ ratios are employed. It should be noted that in the case of a Jacobian free Newton-Krylov approach, for DIRK, the upper bound would still be valid but divided by the number of Newton iterations. Notice that the above cost formula is idealized in the sense that the costs for setting up or applying a non-trivial preconditioning technique were not included. Because the linear system at each stage can *at most* be solved approximatively, due to the use of iterative procedures, traditional Rosenbrock methods cannot be used. In turn, methods able to support arbitrary matrices instead of the true Jacobian are available. These methods are called Rosenbrock W-methods and were first investigated in [9]. Rang and Angermann have derived L-stable Rosenbrock methods of order 3 with 4 stages and this is the method used here [10]. The coefficients of the method in table 1 are for use with the Rosenbrock method in the form of (14) which avoids multiplication by the Jacobian in the rhs. The coefficients were computed using arbitrary precision arithmetics and truncated at the 15^{th} digit. Finally, the linear problem is inverted using a restarted generalized conjugate residual (GCR) Krylov accelerator. The latter is equivalent to GMRES.

4.1 Construction of the Jacobian for Euler

In order to have a practical method, it is necessary to have an efficient evaluation of the Jacobian matrix for the compressible Euler equations. One approach is the

Table 1. L-stable Rosenbrock W-method coefficients with $\gamma = 0.4358665215084590$

$a_{21} =$	2.0000000000000000	$c_{21} =$	-4.588560720558083
$a_{31} =$	1.4192173174557647	$c_{31} =$	-4.184760482319161
$a_{32} =$	-0.2592322116729697	$c_{32} =$	0.285192017355496
$a_{41} =$	4.1847604823191607	$c_{41} =$	-6.368179200128358
$a_{42} =$	-0.2851920173554959	$c_{42} =$	-6.795620944466836
$a_{43} =$	2.2942803602790417	$c_{43} =$	2.870098604331056
$m_1 =$	0.242123807060954	$\hat{m}_1 =$	3.907010534671192
$m_2 =$	-1.223250583904515	$\hat{m}_2 =$	1.118047877820503
$m_3 =$	1.545260255335102	$\hat{m}_3 =$	0.521650232611491
$m_4 =$	0.435866521508459	$\hat{m}_4 =$	0.500000000000000

Jacobian free technique where the action of the Jacobian on a vector is *approximated* using a Gâteau derivative. Since a Rosenbrock-W method is employed a slight error in the Jacobian has no effects in the consistency of the method but could adversely affect the stability. The multiplication of the vector $\underline{\mathrm{V}}_h$ by the Jacobian, frozen at $\underline{\mathrm{U}}_h^n$, of the discretized right hand side of the Euler equations is defined as

$$J\underline{\mathrm{V}}_h = \frac{\partial R_h}{\partial u}\Big|_{\underline{\mathrm{U}}_h^n} \underline{\mathrm{V}}_h = \frac{R_h(\underline{\mathrm{U}}_h^n + \epsilon \underline{\mathrm{V}}_h^n) - R_h(\underline{\mathrm{U}}_h^n)}{\epsilon} + O(\epsilon). \tag{15}$$

5 Low Mach Number Preconditioning

Current compressible flow solvers are not suitable to simulate, without any modifications, flow fields transitioning from incompressible to compressible regimes. The literature on the subject demonstrates that the extension of a compressible solver to the incompressible regime leads to asymptotically wrong solutions at the low Mach limit. By adding time derivatives in an incompressible system it is possible to obtain a clustering of the hyperbolic eigenvalues such that they are all of the same magnitude when compared to one another: see [11] and references therein. However these techniques were developed in order to march the equations towards a steady state. Viozat and collaborators [12,13] have shown how to taylor such an approach to unsteady low Mach number simulations in the finite-volumes case. In what follows we adapt the Russanov flux for the equation proposed in [4] for the low Mach regime. The first step consist into rewriting in entropy variables $\mathrm{W} = (p, u, v, s)^T$ the system using the following *passage* matrices:

$$\frac{\partial \mathrm{W}}{\partial \mathrm{U}} = \begin{pmatrix} c^2 & 0 & 0 & 0 \\ -u/\rho & 1/\rho & 0 & 0 \\ -w/\rho & 0 & 1/\rho & 0 \\ \gamma/\rho & 0 & 0 & \frac{\gamma\, R\, p_{\mathrm{ref}}^{\frac{1-\gamma}{\gamma}}}{p^{1/\gamma}} \end{pmatrix} \quad \text{and} \quad \frac{\partial \mathrm{U}}{\partial \mathrm{W}} = \begin{pmatrix} 1/c^2 & 0\,0 & 0 \\ u/c^2 & \rho\,0 & 0 \\ w/c^2 & 0\;\rho & 0 \\ \frac{1}{\gamma R}\frac{p}{p_{\mathrm{ref}}}^{\frac{1-\gamma}{\gamma}} & 0\,0 & \frac{p_{\mathrm{ref}}}{\gamma R}(\frac{p}{p_{\mathrm{ref}}})^{1/\gamma} \end{pmatrix}$$

where $\mathrm{U} = (\rho, \rho u, \rho w, \rho\theta)^T$ and $s \equiv \log p/\rho^\gamma$. The preconditioning employed is the same as in [13] and consist, in the entropy variables, in a diagonal matrix denoted by $P^{-1}(\mathrm{W}) = \mathrm{diag}\{\beta^2, 1, 1, 1\}$ with $\beta = O(M)$.

$$P(\mathrm{W})\mathrm{W}_t + P(\mathrm{W})\mathbf{G}(\mathrm{W}) \cdot \nabla \mathrm{W} = P(\mathrm{W})\tilde{M}\mathrm{W} \tag{16}$$

where the quasi-linear form of the compressible Euler equations was employed. Transforming back to the conserved variables yields the preconditioning matrix

$$P(\mathrm{U}) \equiv \frac{\partial \mathrm{U}}{\partial \mathrm{W}} P(\mathrm{W}) \frac{\partial \mathrm{W}}{\partial \mathrm{U}}$$

and the corresponding quasi-linear system of conservation laws

$$P(\mathrm{U})\mathrm{U}_t + P(\mathrm{U})\mathbf{A}(\mathrm{U}) \cdot \nabla \mathrm{U} = P(\mathrm{U})\ M\mathrm{U}. \tag{17}$$

The Russanov flux is modified next only in the dissipative term in analogy to what was proposed by [11]. More precisely the maximum eigenvalues are computed with respect to the hyperbolicity matrix $P(\mathrm{U})\mathbf{A}(\mathrm{U}) \cdot \hat{n}$ and multiplied by $P^{-1}(\mathrm{U})$:

$$\mathbf{F}(\mathrm{U}^R, \mathrm{U}^L) \cdot \hat{n} = \frac{1}{2}(\mathbf{F}(\mathrm{U}^R) + \mathbf{F}(\mathrm{U}^L)) \cdot \hat{n} - \frac{1}{2} P^{-1}(\frac{\mathrm{U}^R + \mathrm{U}^L}{2}) D(\mathrm{U}^R, \mathrm{U}^L; \hat{n}) \tag{18}$$

with

$$D(\mathrm{U}^R, \mathrm{U}^L; \hat{n}) = \mathrm{diag}(\max\{|\mathrm{eigen}\{P(\mathrm{U})\mathbf{A}(\mathrm{U}) \cdot \hat{n}\}|\})(\mathrm{U}^R - \mathrm{U}^L).$$

Thus, as remarked in [14] the only modification required to precondition the system are performed in the dissipation matrix. It will later be seen that these modifications lead to lower iteration counts in the iterative solution process when the considered domains are non-hydrostatic.

6 Numerical Experiments

6.1 Rising Smooth Bubble Experiment

We carry an integration using as initial condition a continuous perturbation of the potential temperature in an initially hydrostatically balanced atmosphere. The complete details of the initial condition can be found in [15]. The computational domain consist in a rectangle of dimension $20km$ in the horizontal by $10km$ in the vertical with slip boundary conditions on the top and bottom of the atmosphere and periodic conditions on both ends. The initial perturbation is integrated for a 1000 seconds. The same test is performed with two different values of β. In table 2 the solver is used with the low Mach preconditioning (LM) set with $\beta = 0.4$ while the modification is turned off by using $\beta = 1.0$. The acceleration nearly doubles at large Courant numbers.

Table 2. Effects of low Mach (LM) number preconditioning for the raising bubble experiment. Relative residual tolerance set to Solver 1×10^{-6}, with 7^{th} degree polynomials per element and 16 elements in the horizontal direction and 8 in the vertical.

Time step	Iterations with LM	acceleration	Iterations **without** LM	acceleration
$1.0s$	30	3.2	33	2.8
$2.0s$	36	5.1	45	4.1
$10.0s$	69	13.5	103	9.1
$50.0s$	207	22.7	493	10.2

6.2 Inertia Gravity Waves in a Periodic Channel

An initial perturbation in a periodic channel triggers internal gravity waves that are transported by a background mean flow of $20m/s$ to the right of the domain. The original full definition of the initial condition can be found in [16]. The domain is again a rectangle of dimensions $[300, 10]km$. The translating perturbation is integrated for 3000 seconds. The domain is covered using $[90, 3]$ elements of polynomial of degree 8 leading to a relative mesh spacing of $\Delta x = \Delta z = 500m$ for all simulations. In Fig. 1 left panel, a time step of 12 seconds is employed which is twice the time step used in [16] for a comparable resolution. For a large unphysical time-step, the solution is significantly degraded but stable: Fig. 1 right panel. For time-steps up to 50 seconds which is twice the advective scale $(500/20 = 25)$ the approach produces reasonable solutions.

6.3 Eady Model

The same test as in the previous section is modified by considering an *Eady model*. The forcing term f is set to 0.0001 and the periodic channel is transformed to a very thin shell of dimensions $[6000, 10]km$. The latter is tiled using 600 elements of dimension $1km^2$ each of 7^{th} degree. The model is integrated for 60000 seconds with a resolution of approximatively $1.5km$ in both spatial directions. In this configuration $\beta = 1.0$ lead to faster compute times. The time steps employed in this simulations were ranging from $500s$ to $2000s$ at which an acceleration of

a: b:

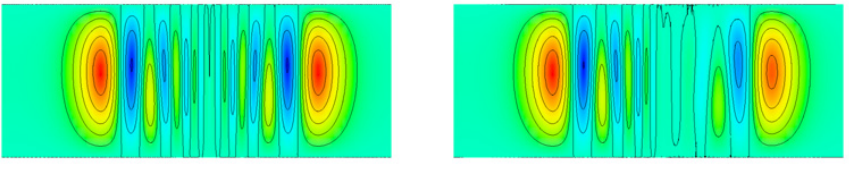


Fig. 1. Inertia gravity wave in a periodic channel using the equations in potential temperature density form. The various time-steps (in alphabetical order) for the third order Rosenbrock W-method are: 12 secs and 100 secs.

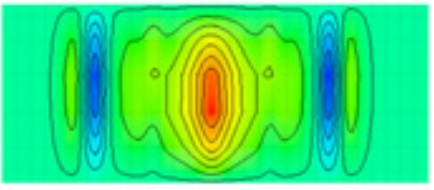


Fig. 2. Inertia gravity wave in a periodic channel in hydrostatic configuration using the Eady model. The various time for the third order Rosenbrock W-method is 500 secs.

45 with respect to explicit was observed. The extreme gains are explained by the exact vertical preconditioning in this case: a consequence of using one element in the vertical direction.

7 Conclusions and Future Directions

In this work it was shown how to discretize the stratified compressible Euler equations for low Mach, nearly incompressible atmospheric simulations consisting in slight perturbation of a hydrostatically balanced state. A low Mach preconditioning of the numerical flux used for yielding asymptotically valid results leads to lower iteration counts for the Krylov accelerator. All simulations result in lower simulation times for simple block Jacobi preconditioning[1]. However, the flux preconditioning needs to take into account the aspect ratio of the domain in order to differentiate between a nonhydrostatic and hydrostatic regimes. Moreover, the numerical results presented show that the nonlinear cycle present in Jacobian-Free Newton Krylov method can be *avoided* and provides a robust framework for unsteady computations. For very thin domains, where the hydrostatic assumption could be employed, the solver shows very good accelerations. If these results are extrapolated to global climate or weather modeling, the corresponding time-steps sizes at 10km resolution would be close to 30 minutes: the time scale employed by most physical parameterizations. Future work will concern the coupling to idealized physics, inclusion of moisture transport, adaptive hybrid non-conforming grids, extension to three spatial dimensions, higher-order Rosenbrock methods and, most importantly, improved preconditioning.

References

1. Cockburn, B., Karniadakis, G.E., Shu, C.W.: Discontinuous Galerkin Methods: Theory, Computation, and Applications. Lecture Notes in Computational Science and Engineering, vol. 11. Springer, New York (2000)

[1] The quality of the non-linear solutions can be readily compared with more traditional approaches at `http://www.mmm.ucar.edu/projects/srnwp_tests/`

2. St-Cyr, A., Thomas, S.J.: Parallel atmospheric modeling with high-order continuous and discontinuous galerkin methods. In: Deane, A., Periaux, J., Ecer, A., Satofuka, N., McDonough, J. (eds.) Parallel Computational Fluid Dynamics 2005: Theory and Applications: Proceedings of the Parallel CFD 2005 Conference, pp. 485–492. Elsevier Science, Amsterdam (2006)
3. Bhanot, G., Dennis, J.M., Edwards, J., Grabowski, W., Gupta, M., Jordan, K., Loft, R.D., Sexton, J., St-Cyr, A., Thomas, S.J., Tufo, H.M., Voran, T., Walkup, R., Wyszogrodzki, A.A.: Early experiences with the 360tf ibm bluegene/l platform. Int. J. Comput. Meth. 5(2), 237–253 (2008)
4. Skamarock, W.C., Klemp, J.B., Dudhia, J., Gill, D.O., Barker, D.M., Wang, W., Powers, J.G.: A description of the advanced research WRF version 2. NCAR Tech. Note TN-468+STR, National Center for Atmospheric Research, 1850 Table Mesa Drive, Boulder, Colorado, 80305, USA (2005) (revised January 2007)
5. Bassi, F., Rebay, S.: High-order accurate discontinuous finite element solution of the 2d euler equations. Journal of Computational Physics 138(2), 251–285 (1997)
6. Giraldo, F.X., Hesthaven, J.S., Warburton, T.: Nodal high-order discontinuous galerkin methods for the spherical shallow water equations. Journal of Computational Physics 181(2), 499–525 (2002)
7. Ronquist, E.M.: Optimal Spectral Element Methods for the Unsteady Three-Dimensional Incom- pressible Navier-Stokes Equations. Ph.D thesis. MIT, Cambridge, MA, USA (1988)
8. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II, 2nd edn. Springer Series in Computational Mathematics, vol. 14. Springer, Heidelberg (1996)
9. Steihaug, T., Wolfbrandt, A.: An attempt to avoid exact jacobian and nonlinear equations in the numerical solution of stiff differential equations. Math. Comp. 33, 521–534 (1979)
10. Rang, J., Angermann, L.: New Rosenbrock W-methods of order 3 for partial differential algebraic equations of index 1. BIT Numerical Mathematics 45, 761–787 (2005)
11. Meister, A.: Asymptotic based preconditioning technique for low Mach number flows. ZAMM 83(1), 3–25 (2003)
12. Viozat, C.: Calcul d'écoulements stationnaires et instationnaires à petit nombre de Mach, et en maillages étirés. Ph.D thesis, Université de Nice-Sophia Antipolis, Nice, France (October 1998)
13. Guillard, H., Viozat, C.: On the behaviour of upwind schemes in the low mach number limit. Comput. Fluids 28, 63–86 (1999)
14. Mavriplis, D.J.: Multigrid strategies for viscous flow solvers on anisotropic unstructured meshes. Journal of Computational Physics 145, 141–165 (1998)
15. Wicker, L.J., Skamarock, W.C.: Time splitting methods for elastic models using forward time schemes. Mon. Wea. Rev. 130, 2088–2097 (2002)
16. Skamarock, W.C., Klemp, J.B.: Efficiency and accuracy of the Klemp-Wilhelmson time-splitting technique. Mon. Wea. Rev. 122, 2623–2630 (1994)

Time Acceleration Methods for Advection on the Cubed Sphere

R.K. Archibald, K.J. Evans, J.B. Drake, and J.B. White III

Oak Ridge National Laboratory, Oak Ridge, TN

Abstract. Climate simulation will not grow to the ultrascale without new algorithms to overcome the scalability barriers blocking existing implementations. Until recently, climate simulations concentrated on the question of whether the climate is changing. The emphasis is now shifting to impact assessments, mitigation and adaptation strategies, and regional details. Such studies will require significant increases in spatial resolution and model complexity while maintaining adequate throughput. The barrier to progress is the resulting decrease in time step without increasing single-thread performance. In this paper we demonstrate how to overcome this time barrier for the first standard test defined for the shallow-water equations on a sphere. This paper explains how combining a multiwavelet discontinuous Galerkin method with exact linear part time-evolution schemes can overcome the time barrier for advection equations on a sphere. The discontinuous Galerkin method is a high-order method that is conservative, flexible, and scalable. The addition of multiwavelets to discontinuous Galerkin provides a hierarchical scale structure that can be exploited to improve computational efficiency in both the spatial and temporal dimensions. Exact linear part time-evolution schemes are explicit schemes that remain stable for implicit-size time steps.

1 Introduction

Large-scale scientific computing has maintained its exponential growth via the ever expanding parallelism while individual processor speeds have begun to stagnate [9]. This trend requires the development of new algorithms that can overcome the time barrier, or effectively scale in spatial resolution while maintaining adequate throughput and accuracy. This paper takes a step towards this goal by demonstrating how the time step for advection equations on a sphere can be significantly increased by using a multiwavelet discontinuous Galerkin method with an exact linear part time-evolution scheme.

The discontinuous Galerkin (DG) method has an elegant and flexible formulation that can provide high-order accurate solutions to complicated models [5,6]. DG is a finite element method that is locally conservative and allows for an element-wise discontinuous solution approximation. DG is a scalable method because numerical information of each element is only passed locally through numerical fluxes to the nearest neighbors. In a set of papers, the DG method

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 253–262, 2009.

was successfully implemented on the sphere for advection models [11] and the shallow water equation [12]. We build on this work by merging multiwavelets with discontinuous Galerkin on the sphere and accelerate the time step by using an exact linear part (ELP) time-evolution scheme.

Multiwavelets are a discontinuous, orthogonal, compactly supported, multi-scale set of functions with vanishing moments that yield high-order *hp*-adaptive approximations of L^2 functions [1]. Combination of multiwavelets with the DG method results in a computationally fast and effective multi-scale adaptive DG method [3]. ELP has been demonstrated to be particularly effective and efficient for multiwavelet-based schemes [2,4] since the operators generated for the ELP method remain sparse in a multiwavelet representation.

This paper is orgainized as follows. In section 2 we introduce the multiwavelet basis and its key features. In section 3 we describe the DG method for the cubed sphere and further demonstrate how multiwavelets are incorporated. Section 4 describes ELP for the multiwavelet DG method. Section 5 demonstrates the time acceleration of advection problems on the cubed sphere. Section 6 ends the paper with a discussion of the results.

2 Multiwavelet Bases

In this section we briefly summarize the important properties of the multiwavelet basis derived and developed in [1] and introduce notation as given in [2]. We begin by defining $\mathbf{V}_n^k$ as a space of piecewise polynomial functions, for $k = 1, 2, \ldots$, and $n = 0, 1, 2, \ldots$, as

$$\mathbf{V}_n^k = \{f : f \in \Pi_k(I_{nl}), \text{ for } l = 0, ..., 2^n - 1, \text{ and } \mathrm{supp}(f) = I_{nl}\}, \tag{1}$$

where $\Pi_k(I_{nl})$ is the space of all polynomials of degree less than k on the interval $I_{nl} = [2^n l, 2^n(l+1)]$. Using this space, we can describe not only multiwavelets, but the solution space that the DG method uses for approximation. The multiwavelet subspace $\mathbf{W}_n^k$, $n = 0, 1, 2, \ldots$, is defined as the orthogonal complement of $\mathbf{V}_n^k$ in $\mathbf{V}_{n+1}^k$, or

$$\mathbf{V}_n^k \oplus \mathbf{W}_n^k = \mathbf{V}_{n+1}^k, \quad \mathbf{W}_n^k \perp \mathbf{V}_n^k. \tag{2}$$

The immediate result of this definition of the multiwavelet subspace is that it splits $\mathbf{V}_n^k$ into $n+1$ orthogonal subspaces of different scales, as

$$\mathbf{V}_n^k = \mathbf{V}_0^k \oplus \mathbf{W}_0^k \oplus \mathbf{W}_1^k \oplus \ldots \oplus \mathbf{W}_{n-1}^k. \tag{3}$$

Given a basis $\phi_0, \ldots, \phi_{k-1}$ of $\mathbf{V}_0^k$, the space $\mathbf{V}_n^k$ is spanned by $2^n k$ functions which are obtained from $\phi_0, \ldots, \phi_{k-1}$ by dilation and translation,

$$\phi_{jl}^n(x) = 2^{n/2}\phi_j(2^n x - l), \quad j = 0, \ldots, k-1, \quad l = 0, \ldots, 2^n - 1. \tag{4}$$

By construction similar properties hold for multiwavelets. If the piecewise polynomial functions $\psi_0, ..., \psi_{k-1}$ form an orthonormal basis for $\mathbf{W}_0^k$, then by dilation and translation the space $\mathbf{W}_n^k$ is spanned by $2^n k$ functions

$$\psi_{jl}^n = 2^{n/2}\psi_j(2^n x - l), \quad j = 0, \ldots, k-1, \quad l = 0, \ldots, 2^n - 1. \tag{5}$$

A function $f \in \mathbf{V}_n^k$ can be represented by the following expansion of scaling functions.

$$f(x) = \sum_{l=0}^{2^n-1} \sum_{j=0}^{k-1} s_{jl}^n \phi_{jl}^n(x), \tag{6}$$

where the coefficients s_{jl}^n are computed as

$$s_{jl}^n = \int_{2^{-n}l}^{2^{-n}(l+1)} f(x)\phi_{jl}^n(x)dx. \tag{7}$$

The decomposition of $f(x)$ has an equivalent multiwavelet expansion given by

$$f(x) = \sum_{j=0}^{k-1}(s_{j0}^0\phi_j(x) + \sum_{m=0}^{n-1}\sum_{l=0}^{2^m-1} d_{jl}^m\psi_{jl}^m(x)), \tag{8}$$

with the coefficients

$$d_{jl}^m = \int_{2^{-n}l}^{2^{-n}(l+1)} f(x)\psi_{jl}^n(x)dx. \tag{9}$$

It is demonstrated in [1] how fast transforms between (6) and (8) can be developed using two-scale difference equations. Specifically, expansion coefficients of multiwavelets with k vanishing moments can be constructed on consecutive levels m and $m+1$ through repeated application of

$$\begin{aligned} s_{jl}^m &= \sum_{j=0}^{k-1}(h_{ij}^{(0)}s_{j,2l}^{m+1} + h_{ij}^{(1)}s_{j,2l+1}^{m+1}), \\ d_{jl}^m &= \sum_{j=0}^{k-1}(g_{ij}^{(0)}s_{j,2l}^{m+1} + g_{ij}^{(1)}s_{j,2l+1}^{m+1}), \end{aligned} \tag{10}$$

using the scaling coefficients $h_{ij}^{(0)}$ and $g_{ij}^{(0)}$ for $i, j = 0, \ldots, k-1$. The inverse operation that takes expansion coefficients of (8) to (6) is given by

$$\begin{aligned} s_{j,2l}^{m+1} &= \sum_{j=0}^{k-1}(h_{ji}^{(0)}s_{j,l}^m + g_{ji}^{(0)}d_{jl}^m), \\ d_{j,2l+1}^{m+1} &= \sum_{j=0}^{k-1}(h_{ji}^{(1)}s_{j,l}^m + g_{ji}^{(1)}d_{j,l}^m), \end{aligned} \tag{11}$$

for the scaling coefficients $h_{ij}^{(0)}$ and $g_{ij}^{(0)}$, for $i, j = 0, \ldots, k-1$.

The total number of expansion coefficients in (6) and (8) are the same, but the number of *significant* expansion coefficients for a given error tolerance level ϵ will be different. A benefit of using the multiwavelet expansion (8) is that much-fewer significant expansion coefficients are generally needed. A result of this property when multiwavelets are used in DG methods is an increase in computational speed and efficiency [1]. In this paper we use hard thresholding to eliminate non-significant expansion coefficients.

3 Multiwavelet Discontinuous Galerkin Method on the Cube Sphere

In this section we begin by describing the multiwavelet DG [1] method in two dimensions and finish by demonstrating how this method can be used with the cube-sphere geometry to model equations on the sphere.

Consider the two-dimensional scalar nonlinear conservation law

$$u_t + \nabla \cdot f(u) = 0, \text{ in } [0,1]^2 \times [0,T]. \tag{12}$$

We restrict our attention to uniform Cartesian meshes since they provide the most natural representation for multiwavelets; other mesh choices are possible but the implementation becomes more challenging [7]. Given a fixed order $k \geq 0$ and resolution $n \geq 0$, variational formulation of the DG method is derived by multiplying (12) by the test functions $\phi_{jl} \in \mathbf{V}_n^k$ and integrating to obtain

$$\begin{aligned}\int_{I_{n\ell}} \int_{I_{nl}} \frac{\partial u}{\partial t} \phi_{jl}^n(x) \phi_{\jmath\ell}^n(y) dxdy = & \int_{I_{n\ell}} \int_{I_{nl}} f(u) \frac{\partial \phi_{jl}^n(x)}{\partial x} \phi_{\jmath\ell}^n(y) dxdy \\ & + \int_{I_{n\ell}} \int_{I_{nl}} f(u) \phi_{jl}^n(x) \frac{\partial \phi_{\jmath\ell}^n(y)}{\partial y} dxdy \\ & - \int_{\partial[I_{n\ell} \times I_{nl}]} f(u) \cdot \mathbf{n}\ \phi_{jl}^n(x) \phi_{\jmath\ell}^n(y) ds, \end{aligned} \tag{13}$$

for $j, \jmath = 0, \ldots, k-1$ and $l, \ell = 0, 1, \ldots, 2^n - 1$, where $\mathbf{n}$ is the outward-facing unit normal vector on the element boundary $\partial[I_{n\ell} \times I_{nl}]$. Consider the following two-dimensional multiwavelet expansion.

$$u_h(x,y,t) = \sum d_{jl,\jmath\ell}^{m,\mu} \psi_{jl}^m(x) \psi_{\jmath\ell}^\mu(y), \tag{14}$$

with summation taken over $j, \jmath = 0, \ldots, k-1$ and $m, \mu = -1, 0, \ldots, n-1$ and $l = 0, 1, \ldots, \min(0, 2^m - 1)$ and $\ell = 0, 1, \ldots, \min(0, 2^\mu - 1)$, where notation is condensed by defining $\psi_{j0}^{-1}(\cdot) \equiv \phi_j(\cdot)$, for $j = 0, \ldots, k-1$. The numerical multiwavelet DG scheme supplants the test functions (13) with multiwavelets and solves

$$\begin{aligned}\int_{I_{n\ell}} \int_{I_{nl}} \frac{\partial u_h}{\partial t} \psi_{jl}^n(x) \psi_{\jmath\ell}^n(y) dxdy = & \int_{I_{n\ell}} \int_{I_{nl}} f(u_h) \frac{\partial \psi_{jl}^n(x)}{\partial x} \psi_{\jmath\ell}^n(y) dxdy \\ & + \int_{I_{n\ell}} \int_{I_{nl}} f(u_h) \psi_{jl}^n(x) \frac{\partial \psi_{\jmath\ell}^n(y)}{\partial y} dxdy \\ & - \int_{\partial[I_{n\ell} \times I_{nl}]} \hat{f}(u_h) \cdot \mathbf{n}\ \psi_{jl}^n(x) \psi_{\jmath\ell}^n(y) ds, \end{aligned} \tag{15}$$

where $\hat{f}(u_h)$ is a monotone numerical flux, the focal point for the only communication between elements. Throughout this paper we use the well known simple

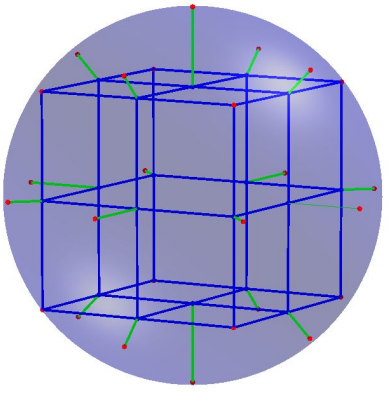


Fig. 1. The cubed-sphere gridding technique projects the red points on the sphere along radial lines to the blue points on the inscribed cube face

Lax-Friedrichs flux [10] and Gauss-Lobatto quadrature for integration. An explicit solution of (15) results directly from the orthogonality of multiwavelets, where

$$\int_{I_{n\ell}} \int_{I_{nl}} \frac{\partial u_h}{\partial t} \psi_{jl}^n(x) \psi_{j\ell}^n(y) dx dy = \frac{\partial d_{jl,j\ell}^{m,\mu}}{\partial t}$$

for all index values given previously.

The cubed sphere, first developed in [13], has proven to be a particularly useful gridding technique for solving partial differential equations on the sphere [11,12,14]. Figure 1 depicts the cubed sphere, where the transformation between the inscribed cube and the sphere is determined by the gnomonic (center) projection from the sphere to each face of the cube. DG is well-suited for this type of gridding [11,12], since each face can be solved as a separate two-dimensional problem, with faces communicating with each other as boundary conditions.

4 Time Discretization

We use a method of time stepping that has been demonstrated to be particularly effective and efficient for multiwavelet schemes [2,4]. The idea behind the development of these schemes, as it is related to this research, is to convert differential equations of the form

$$u_t = \mathcal{L}u + \mathcal{N}(u), \tag{16}$$

where the system is split into a linear operator $\mathcal{L}$ and nonlinear operator $\mathcal{N}$, into the equivalent integral equation,

$$u(t) = e^{t\mathcal{L}} u_0 + \int_0^t e^{(t-\tau)\mathcal{L}} \mathcal{N}(u) d\tau. \tag{17}$$

The multiwavelet basis allows fast scaling and squaring methods that produce sparse and highly accurate approximations to the exponential linear operator. These time-stepping schemes are therefore called *exact linear part* (ELP) schemes.

This paper focuses on linear advection equations on the sphere, and therefore we will only discuss how to approximate the exponential operator $e^{t\mathcal{L}}$. Suppose we are given the matrix $\mathcal{L}$ and an error tolerance ϵ; the scaling and squaring method that approximates the exponential linear operator is as follows.

1. Compute the exponent j such that $t\|\mathcal{L}\|_2/2^j < \epsilon$.
2. Compute the approximation $e^{t\mathcal{L}/2^j} = \mathcal{I} + t\|\mathcal{L}\|_2/2^j$.
3. $e^{t\mathcal{L}/2^j}$ is squared j times to obtain $e^{t\mathcal{L}}$.

Sparsity is maintained by truncating to the error tolerance at each step.

5 Numerical Results

In this section we consider the following problem of advection on the sphere, a problem that has specific importance to the development of climate models.

Example 1. Given the advecting field h, the equation for advection in flux form is

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{v}) = 0. \tag{18}$$

The first test in the standard suit developed by the climate modeling community [15] is to solve (18) on the surface of a sphere, with initial conditions given in spherical coordinates as

$$h\Big(r(\lambda,\theta)\Big) = \begin{cases} \frac{h_0}{2}(1+\cos(\frac{\pi r}{R})) & \text{if } r < R, \\ 0 & \text{otherwise,} \end{cases} \tag{19}$$

for $r(\lambda,\theta) = a\arccos(\sin(\theta_c)\sin(\theta) + \cos(\theta_c)\cos(\theta)\cos(\lambda-\lambda_c))$ and advecting wind

$$\mathbf{v} = u_0 \begin{pmatrix} cos(\theta)cos(\alpha) + sin(\theta)cos(\lambda)sin(\alpha) \\ -sin(\lambda)sin(\alpha) \end{pmatrix}. \tag{20}$$

Here the parameters are set to $a = 6.37122 \times 10^6$m, $h_0 = 1000$m, $(\lambda_c, \theta_c) = (\frac{3\pi}{2}, 0)$, $R = \frac{a}{3}$, $u_0 = \frac{2\pi a}{12 \text{ days}}$, and $\alpha = \frac{\pi}{4}$. We note that this choice of α represents a particularly difficult problem, since the advecting cosine bell passes through four corners and along two edges of the cubed-sphere grid during each full revolution.

Along with the *Cosine bell* initial conditions (19), we will also consider the so called *Gaussian hill* initial conditions,

$$h\Big(r(\lambda,\theta)\Big) = h_0 e^{-\frac{r}{\rho^2}}, \tag{21}$$

for $\rho = 2500$ km.

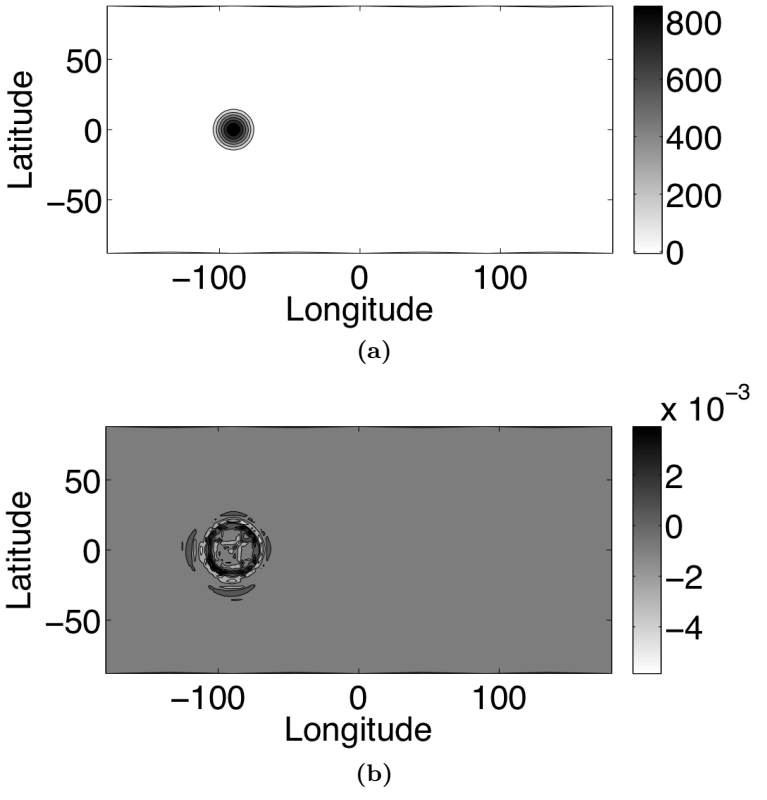


Fig. 2. (a) Final multiwavelet DG solution of Example 1 for *Cosine bell* initial conditions. (b) Relative error after one complete revolution, with $N = 16$, $k = 3$ and CFL = 18.2.

Throughout this section we will use

$$\text{CFL} = \frac{u_0 \Delta t}{\Delta x} \quad \text{and} \quad N_e = 6N^2, \tag{22}$$

where Δt is the time step, N_e is the total number of elements on each cube face, and $\Delta x = \frac{1}{N}$.

Figure 2 depicts the multiwavelet DG solution and relative error of Example 1 for *Cosine bell* initial conditions, with $N = 16$, $k = 3$ and CFL = 18.2. It can be seen that using ELP time stepping provides a stable solution for time steps that significantly exceed the CFL requirement for explicit methods. Figure 3 depicts the same multiwavelet DG solution and relative error of Example 1 for *Gaussian hill* initial conditions. The difference between the multiwavelet DG solution and the exact solution is no more than a fraction of a percent for each initial condition and is considerable better for *Gaussian hill* initial conditions due to the increased smoothness of this initial condition.

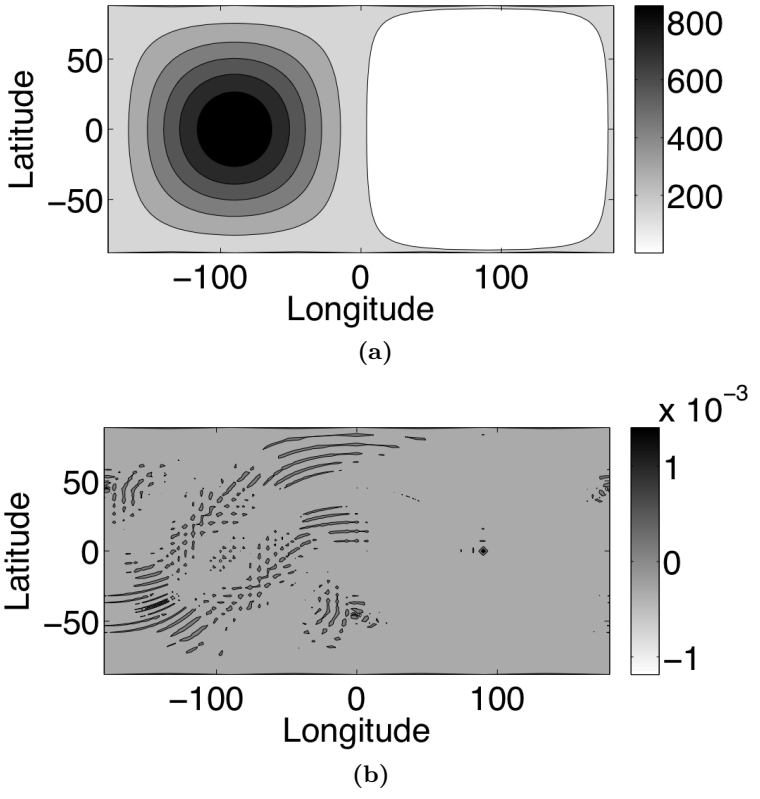


Fig. 3. (a) Final multiwavelet DG solution of Example 1 for *Gaussian hill* initial conditions. (b) Relative error after one complete revolution, with $N = 16$, $k = 3$ and CFL = 18.2.

Table 1 gives more-detailed insight into the properties of multiwavelet DG. We compare fourth order in time Runge-Kutta time stepping (RK4) [8] to ELP. Since Example 1 is linear, we can convert the Runge-Kutta method into an equivalent matrix operation. Each time step for the ELP method also consists of one matrix operation, and therefore we use the number of non-zero elements, N_z, in each time-evolution matrix to give a measure of the computational effort for each time step. Our first observation from Table 1 is that for both types of initial conditions the L_2 error and order of convergence is comparable for each time-stepping method and CFL number. We note that the convergence rates and errors are similar to the results published in [11] for the same problem, with cosine bell initial condition, using a DG method with CFL = 0.1 and a third-order Runge-Kutta method. We report that in this study CFL > 0.35 resulted in instability for the RK4 method. Finally, it can be seen that the ELP method can significantly increase the time step while preserving accuracy. ELP time stepping provided a sixteen-fold acceleration of Runge-Kutta with no significant increase in the number of nonzero elements in the time-evolution matrixes.

Table 1. Convergence rates for Example 1 using RK4 and ELP time stepping for the multiwavelet DG method with order $k = 3$ and drop tolerance $\epsilon = 10^{-4}$ for the ELP with CFL= 4.8 and $\epsilon = 10^{-5}$ otherwise. The number of non-zero elements for each operator is give by N_z.

	RK4 (CFL = 0.3)			ELP (CFL = 4.8)			ELP (CFL = 18.2)		
N	L_2 error	Order	N_z	L_2 error	Order	N_z	L_2 error	Order	N_z
cosine bell									
4	1.98e-1	-	5.7e5	1.98e-1	-	5.9e5	1.96e-1	-	1.5e6
8	4.04e-2	2.30	2.4e6	4.18e-2	2.25	2.5e6	4.11e-2	2.26	8.2e6
16	7.53e-3	2.42	9.9e6	7.61e-3	2.46	1.0e7	7.71e-3	2.14	3.4e7
Gaussian hill									
4	2.0e-2	-	5.7e5	2.01e-2	-	5.9e5	2.02e-2	-	1.5e6
8	3.04e-3	2.72	2.4e6	3.06e-3	2.72	2.5e6	3.08e-3	2.72	8.2e6
16	3.6e-4	3.09	9.9e6	3.62e-4	3.08	1.0e7	3.63e-4	3.08	3.4e7

Also, a sixty-fold time acceleration was achieved at the cost of a three-fold increase in the number of nonzero elements.

6 Conclusions

This research has demonstrated that significant increases in time-step length are possible for advection problems on the cubed sphere by using an ELP multiwavelet DG method as compared to DG. A sixty-fold increase in time step is achieved for the first test in the standard suit developed by the climate modeling community [15] in the most-challenging advection direction for the cubed-sphere geometry. The cost of this time acceleration is a three-fold increase in the number of spatial calculations. This penalty is small relative to the gain in time acceleration and is desirable because spatial operations offer better opportunities for parallelization.

Acknowledgments

This research has been sponsored by the Laboratory Research and Development Program of Oak Ridge National Laboratory (ORNL), managed by UT-Battelle, LLC for the U.S. Department of Energy under Contract No. DE-AC05-00OR22725. Accordingly, the U.S. Government retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

References

1. Alpert, B.: A class of bases in L^2 for the sparse representation of integral operators. SIAM J. Math. Anal. 24(1), 246 (1993)
2. Alpert, B., Beylkin, G., Gines, D., Vozovoi, L.: Adaptive solution of partial differential equations in multiwavelet bases. Journal of Computational Physics 182(1), 149 (2002)

3. Archibald, Fann, Shelton: Adaptive Discontinuous Galerkin Methods in Multiwavelets Bases. Journal of Scientific Computing (2008) (submitted)
4. Beylkin, G., Keiser, J.M., Vozovoi, L.: A new class of stable time discretization schemes for the solution of nonlinear PDEs. Journal of Computational Physics 147, 362 (1998)
5. Cockburn, B., Shu, C.W.: The local discontinuous Galerkin method for time-dependent convection diffusion systems. SIAM Journal on Numerical Analysis 35, 2440 (1998)
6. Cockburn, B., Shu, C.W.: Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. Journal of Scientific Computing 16(3), 173 (2001)
7. Coult, N.: Introduction to Discontinuous Wavelets. In: Cockburn, B., Karniadakis, G.E., Shu, C.-W. (eds.) Discontinuous Galerkin Methods: Theory Computation and Applications. Springer, Heidelberg (2000)
8. Davis, P.J., Polonsky, I.: Numerical Interpretation, Differentiation, and Integration. In: Abramowitz, M., Stegun, I. (eds.) Handbook of Mathematical Functions. Dover (1972)
9. Drake, Jones, Vertenstein, White III, Worley: Software Design for Petascale Climate Science. In: Bader, D. (ed.) Petascale Computing: Algorithms and Applications. Chapman & Hall/CRC, Boca Raton (2008)
10. LeVeque, R.J.: Numerical Methods for Conservation Laws. Birkhauser Verlag, Basel (1990)
11. Nair, R.D., Thomas, S.J., Loft, R.D.: A discontinuous Galerkin transport scheme on the cubed sphere. Monthly Weather Review 133(4), 814 (2005)
12. Nair, R.D., Thomas, S.J., Loft, R.D.: A discontinuous Galerkin global shallow water model. Monthly Weather Review 133(4), 876 (2005)
13. Sadourny, R.: Conservative Finite-Difference Approximations of the Primitive Equations on Quasi-Uniform Spherical Grids. Monthly Weather Review 100(2), 136–144 (1972)
14. Taylor, M.A., Tribbia, J.J., Iskandrani, M.: The spectral element method for the shallow water equations on the sphere. Journal of Computational Physics 130, 92–108 (1997)
15. Williamson, D.L., Hack, J.J., Jakob, R., Swarztrauber, P.N., Drake, J.B.: A standard test set for numerical approximations to the shallow water equations in spherical geometry. Journal of Computational Physics 102, 211 (1992)

Comparison of Traditional and Novel Discretization Methods for Advection Models in Numerical Weather Prediction

Sean Crowell[1], Dustin Williams[1], Catherine Mavriplis[2], and Louis Wicker[3]

[1] University of Oklahoma, USA
[2] University of Ottawa, Canada
[3] National Severe Storms Laboratory, USA
scrowell@ou.edu, vanillaice@ou.edu,
Catherine.Mavriplis@uottawa.ca, Louis.Wicker@noaa.gov

Abstract. Numerical Weather Prediction has been dominated by low order finite difference methodology over many years. The advent of high performance computers and the development of high order methods over the last two decades point to a need to investigate the use of more advanced numerical techniques in this field. Domain decomposable high order methods such as spectral element and discontinuous Galerkin, while generally more expensive (except perhaps in the context of high performance computing), exhibit faster convergence to high accuracy solutions and can locally resolve highly nonlinear phenomena. This paper presents comparisons of CPU time, number of degrees of freedom and overall behavior of solutions for finite difference, spectral difference and discontinuous Galerkin methods on two model advection problems. In particular, spectral differencing is investigated as an alternative to spectral-based methods which exhibit stringent explicit time step requirements.

Keywords: Numerical weather prediction, spectral differencing, discontinuous Galerkin, advection.

1 Introduction

The science of meteorology relies heavily on numerical simulations of the equations that govern the atmosphere. These computations combine observational data and scientific theory to enhance understanding of complex meteorological phenomena. The aim of Numerical Weather Prediction (NWP) is to diagnose the state of the atmosphere and rapidly deliver a prognosis for its future state. Hence the importance of using highly accurate as well as efficient numerical techniques cannot be overstated. While it is true that the current challenges in NWP are associated with parameterization of sub-grid scale processes, whose cost eclipses that of the underlying dynamics, parameterization accuracy in NWP applications depends on the accurate transport of the Reynolds averaged variables to the correct locations at the correct time with the correct structure. Optimizing the accuracy by using higher order schemes involving fewer degrees of freedom can engender significant cost savings in the calculation of these many physical processes.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 263–272, 2009.

Finite difference (FD) numerical methods have been very successful in meteorological modeling. They are easily formulated, well documented and studied, and foster an intuitive computer programming implementation. These methods are intrinsic components of most of today's state-of-the-art dynamical cores in research and operational models. Simply put, they are efficient, comfortable, and proven.

Classic spectral numerical methods have been used in the direct numerical simulation of turbulence and in the arena of global climate modeling. These methods provide exponential convergence, low diffusive and dispersive errors, and have been proven to be more efficient than FD methods when high accuracy or long time integration is required [1]. These methods, however, have not been used extensively in NWP, for which computational efficiency is arguably as important as accuracy. For the same computation, spectral methods require more operations, and often impose more severe time step restrictions, than FD methods. The global property of these methods, which enables their high accuracy, can also destroy their ability to be decomposed in parallel computing architectures.

Element-based spectral (EBS) methods, developed in the last 25 years, are a relatively new approach to numerical modeling. These methods feature the desired localization of FD stencils, as a domain is discretized into smaller sub-domains, or elements, in which local approximations are made. They also provide the formal accuracy and wave propagation properties of classic spectral methods. Of potentially greater importance, EBS methods provide the seemingly ideal framework to exploit high performance computing architectures of the present and future. While these methods are promising, and have been used extensively in engineering, they remain relatively unexplored in NWP, although several on-going efforts, *e.g.* [2,3,4], in global (climate) atmospheric science have proven them worthy of pursuit.

Our work seeks to quantify the differences between FD and EBS methods in terms of accuracy and computational efficiency. The intent is not necessarily to prove one method superior to another, but rather to elucidate the strengths, weaknesses, and behaviors of the methods in modeling meteorological phenomena. It is hoped that a greater understanding of EBS methods will help identify their potential use in meteorological codes. This paper will focus on the introduction of the spectral differencing method as an alternative to FD and Galerkin-based EBS methods.

2 Discretization Methods

With the promise of higher accuracy and potentially corresponding high efficiency, we investigate EBS methods for simple NWP advection models. High order EBS methods rely on spectral decomposition of the solution on sub-domains or elements in terms of high order orthogonal polynomials with corresponding high order quadrature-rule points. The distribution of these points is non-equal in space, necessitating stricter time step requirements in explicit schemes. This single issue has damped the enthusiasm of NWP investigators to consider high order methods. Spectral elements (SE) and discontinuous Galerkin (DG) are two such methods that show promise for NWP. However, the global continuity and the only globally conservative feature of SE also present drawbacks for NWP. DG, on the other hand, provides local conservation and greater flexibility for parallel computing through the calculation of fluxes at element interfaces. Spectral differencing (SD) presents itself as a perhaps viable

alternative to all of these methods: it combines the high order advantages of EBS methods with the collocation approach of FD methods. Furthermore, through recent work, it has been found that the choice of the collocation points can be changed with only slight effect on accuracy, thereby alleviating the restrictions of small time steps. We present here a short description of the SD method and provide a stability analysis.

2.1 Spectral Differencing

The spectral difference method (SD) was first proposed by Kopriva and Kolias [5] under the name "conservative staggered-grid Chebyshev multidomain method." The authors used the method to solve the Euler equations for a number of compressible flows, for varying geometries. Like DG, SD was developed, in part, as a way to discretize and solve problems involving complex geometries. Similar to FD, the staggered grid was constructed to ensure conservation and remove the solution points from boundaries, thus alleviating problems with sub-domain corners and boundary conditions. See the grid in Fig. 1a. The name "spectral difference" was coined by Liu *et al.* [6], who reprised a slight variance on the original Chebyshev multi-domain method. SD was sought after as an alternative to spectral methods that require quadrature.

Van den Abeele *et al.* [7], recently provided an analysis of the stability of SD, and found that the method can be unstable under certain conditions. The method to stabilize the scheme proposed in [7] is to "cluster" the interior flux points toward the element edges in a manner proven to eliminate the positive real Fourier footprint of the spatial discretization, as shown in Fig. 1b). This paper also showed that the numerical dispersion relation is independent of the position of the solution points, which means that the solution points can be placed arbitrarily without affecting the stability or dispersive properties of the method. This leads to an important gain in the efficiency of the method. The solution points can be placed at flux points thereby reducing the amount of interpolation required when the solution polynomial is interpolated to the flux grid. When a solution point is collocated with a flux point, no interpolation to

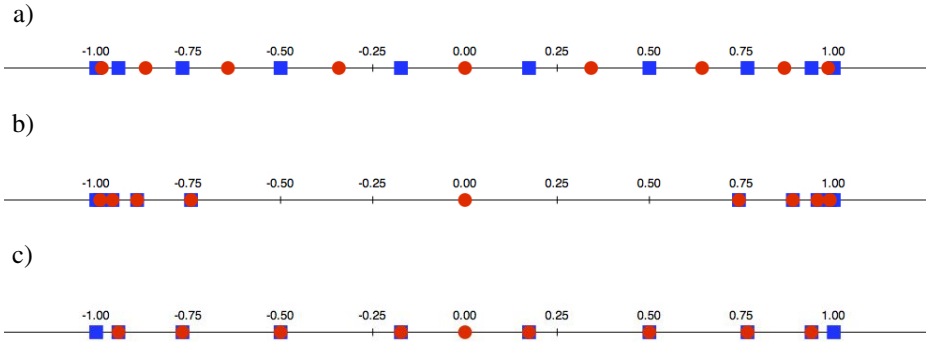


Fig. 1. The spectral difference collocation staggered grids of a) Kopriva and Kolias [5] b) Van den Abeele *et al.* [7] and c) the present study for an 8th order approximation. The red circles represent the grid on which the solution is defined over one element. The blue squares represent the grid on which the flux is computed. The red circles overlaid on blue squares represent flux points at which solution points have been placed.

that flux point is needed; the value of the solution is simply transferred to the flux point. If the solution points are to remain off the element boundaries, one of the desirable properties of the staggered grid, then they are only overlaid at interior flux points. The SD formulation used in this study is a combination of the original method of Kopriva and Kolias [5] and the new method of Van den Abeele *et al.* [7]. The Gauss Lobatto Chebyshev quadrature nodes are used as the flux points, as in [5], but the solution point independence demonstrated by [7] is exploited as shown in Fig. 1c). Thus, both the severe time step restriction of the "stretched" flux points of [7], and the need to interpolate the solution polynomial to all flux points as in [5], are avoided.

The stability analysis is a variance on the traditional von Neumann wave analysis for linear problems. The von Neumann method represents the discretized solution at time n by a finite Fourier series,

$$\phi_j^{\,n} = \sum_{k=-N}^{N} a_k^{\,n} \exp(ikj\Delta x), \tag{1}$$

and examines the stability of an arbitrary Fourier mode, $\exp(ikj\Delta x)$. Finite Fourier series have the property that individual modes are eigenfunctions of linear FD operators [8] so the solution at time $n+1$ can be represented as

$$\phi_j^{\,n+1} = A_k \exp(ikj\Delta x), \tag{2}$$

where A_k is a complex constant known as the "amplification factor." The amplification factor does not vary from time step to time step, so the stability of each Fourier mode is determined by the modulus of its amplification factor. The von Neumann stability condition

$$|A_k| \le 1, \tag{3}$$

ensures that all resolvable Fourier modes are stable. The solution, Eqn. (1), will then be guaranteed to be stable since it is a linear combination of Fourier modes. The present stability analysis for the one-dimensional linear advection equation, similar to that of [7], represents the operators involved in a SD time step by a single matrix. If the vector of solution values at time n is u, then solution at time $n+1$ is

$$u^{n+1} = Bu^n, \tag{4}$$

where B is the operator matrix constructed by the SD interpolation, flux differentiation, and 3rd order Runge-Kutta time integration. Van den Abeele *et al.* [7] determined that for linear problems, an upwind flux is unstable for approximation order two and higher when the Gauss Lobatto Chebyshev quadrature nodes are used as the flux points. This analysis was performed for the SD spatial discretization and an Euler forward step in time. The present analysis, though for a 3rd order Runge-Kutta time integration, verifies these findings. Fig. 2 shows plots of the maximum eigenvalue versus the Courant number, $c\Delta t/\Delta x$, for $N = 4$, 8 approximations using a Lax-Friedrichs (corresponding to upwind in this case) flux. Within the range of reasonable Courant numbers, both plots clearly show an area for which the maximum eigenvalue exceeds one, and is therefore unstable. Note that, for a given order of approximation, not all Courant numbers are unstable. The range of unstable Courant numbers decreases with increasing order of approximation, as does the magnitude of the maximum eigenvalues. The 8th order

instability is particularly small. Using a sine wave initial condition, a wave speed of unity, 10 elements, and a Courant number of 0.1, the 8th order SD approximation took approximately 12,700,000 time steps before the solution demonstrated instability!

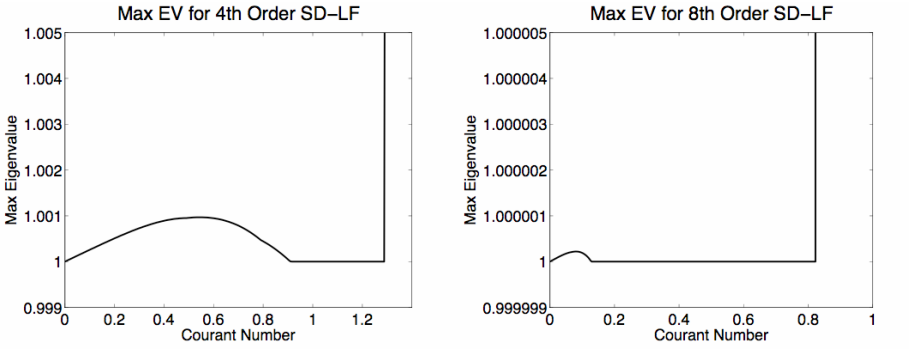


Fig. 2. Maximum eigenvalue of the amplification matrix of the spectral differencing operator with Lax-Friedrichs flux calculation as a function of Courant number for 4^{th} (left) and 8^{th} (right) order basis polynomials. Notice the difference in scale.

Van den Abeele *et al.* [7] did not address the use of any other numerical flux for linear problems, perhaps for good reason. The upwind flux for linear problems makes the most sense physically. When all waves travel with the same speed and in the same direction, all information is being propagated from only the upwind direction.

Nonetheless, in order to fully evaluate SD, we consider a central flux for the one-dimensional linear advection equation using the Gauss Lobatto Chebyshev flux points. It is found that the central flux is stable: for all orders of approximation investigated (N = 2, … 10), the maximum eigenvalue never exceeds one within the range of reasonable Courant numbers. In particular, for N=4, the method is stable for Courant numbers < 1.096, and, for N=8, for Courant < 0.605. As with FD schemes, the maximum stable Courant number decreases with increasing order of approximation, but asymptotes in the vicinity of 8th order schemes. Both central and Lax-Friedrichs (L-F) fluxes were investigated in the numerical tests. More details on the stability analysis are available in Williams' thesis [9].

With these stability and time step restrictions in mind, a new SD grid is adopted that places the flux points at the Gauss Lobatto Chebyshev quadrature nodes and overlays the solution points on flux points, as was suggested by [7]. Fig. 1c) shows the grid for 8th order approximations. The new SD grid (Fig. 1c)) is much more regularly spaced than that of [7] especially for orders of 6 and higher. The time step restriction for the new SD grid is greatly reduced from that imposed by the grid in [7].

2.2 Finite Differencing

The SD method will be compared with standard finite differencing (see Durran [8] for example) of orders 2, 4, and 6, since these are the most commonly used NWP FD stencils. The equations will be solved in flux form to match the DG and SD formulations

and because the flux form is useful to ensure conservation in NWP. The spatial derivatives are computed for the fluxes, which are functions of the vector solution values. The staggered grid interpolation of Shchepetkin and McWilliams [10] is employed. Details are given in [9].

2.3 Discontinuous Galerkin Method

The DG method is a weighted residual technique requiring flux corrections at discontinuous element interfaces as a result of integration by parts of the weak form to ensure conservation. The present implementation follows derivations of Cockburn and Shu [11] with Lax-Friedrichs flux calculation. Legendre polynomial basis functions are used along with Gauss Lobatto Legendre quadrature. Details are given in [9]. Numerical results for DG, SD and FD discretizations are compared in the next section.

3 Numerical Tests

In order to verify the stability results of the preceding sections and to examine other properties of the schemes, such as dissipation and dispersion, several test problems were solved using each of the three general frameworks: FD, SD and DG. In the case of SD and DG, the basis polynomial order was varied, as well as the number of elements. In addition to qualitative comparisons, plots of root-mean-square error (RMSE) versus CPU time and number of degrees of freedom were compiled. Finally, we measured the smallest number of degrees of freedom required for each method (for a particular degree) to achieve a certain order of magnitude of RMSE.

3.1 1D Constant Speed Advection

Constant speed advection is governed by the wave equation: $\frac{\partial u}{\partial t} - c\frac{\partial u}{\partial x} = 0$.

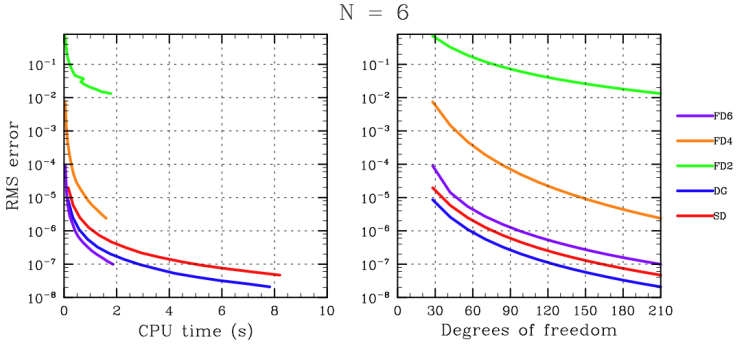


Fig. 3. RMSE in wave equation sine wave solution against CPU time (left) and total number of degrees of freedom (right) for 2nd, 4th and 6th order FD, 6th order DG and 6th order SD

In this case we consider the advection of two initial conditions (one smooth (sine wave) and one nonsmooth (square wave)) through the domain, with periodic boundary conditions. The solutions by the three methods after the sine wave has passed through the domain twenty times (for $c = 1$) are indistinguishable and are therefore not shown. Fig. 3 shows RMSE versus CPU time and the total number of degrees of freedom, for DG and SD taken to be the total number of (solution) nodes. Table 1 shows how many degrees of freedom (nodes) are needed to attain a particular order of magnitude of accuracy. In all cases we compare 6^{th} order DG-LF and SD with central flux against 2^{nd}, 4^{th} and 6^{th} order FD. While higher order will yield better results, we have restricted this presentation to orders typically used in NWP.

Table 1. Degrees of freedom required to achieve an approximate RMSE for the wave equation sine wave solution using FD, DG and SD (2^{nd}, 4^{th} and 6^{th} order)

Order	2^{nd}			4^{th}			6^{th}		
RMSE	SD	DG	FD	SD	DG	FD	SD	DG	FD
10^{-1}	18	24	78	10	10	20	7	7	14
10^{-2}	27	39	246	10	15	30	7	14	14
10^{-3}	45	69	768	20	20	50	14	14	21
10^{-4}	81	123	2421	25	30	85	21	21	35
10^{-5}	141	231	~8000	40	50	150	42	28	49

3.2 2D Nonconstant Advection – Smolarkiewicz Deformational Flow

In this problem, a temperature bubble is placed into a field of vortices, that then advect the bubble along circular streamlines, following the advection equation:

$$\frac{\partial T}{\partial t}+\frac{\partial}{\partial x}(uT)+\frac{\partial}{\partial y}(vT)=0 \text{ with } \begin{matrix} u = A\sin(kx)\sin(ky) \\ v = A\cos(kx)\cos(ky) \end{matrix} \text{ and } \begin{matrix} A = 8 \\ k = \dfrac{4\pi}{100} \end{matrix} .$$

There is a "breaking time" at $t_b = 263.76$ where the gradients become infinitely large. We will examine solutions at $t = t_b / 5 = 52.752$ when the solution exhibits some roll-up and some very fine structures. This problem is relevant to meteorology because it simulates the effects of closed vortices on warm parcels of air. More details can be found in [9], [12] and [13] which provides the exact solution.

Fig. 6 presents the exact solution as well as 6^{th} order FD, SD and DG solutions. The SD results for the 2D Smolarkiewicz flow do not match what we would expect from our 1D advection stability analysis. That is, the central flux case is unstable even for a Courant number of 0.1, while the Lax-Friedrichs flux exhibits no instability at the times investigated. Further, the SD-LF exhibits better errors than the other methods for this test. This is most likely due to the change in character of the Lax-Friedrichs formulation from upwind in the 1D advection case to something more sophisticated in the 2D quasilinear case. The global RMSE results of the DG and SD

solutions are poor, due to severe overshoots near the corners and sharp features in the solution. [Note that while it is possible to adjust the grid and/or use a filter to improve these results, we wanted to present a straightforward comparison of the three methods in this paper, using similar resolution, in order to elucidate the strengths and weaknesses of the respective methods.] However, a spatial map of the error reveals DG to be the best solution overall when matching CPU time and approximately similar number of degrees of freedom, whereas the FD solution is smeared as expected due to the higher diffusion and dispersion errors.

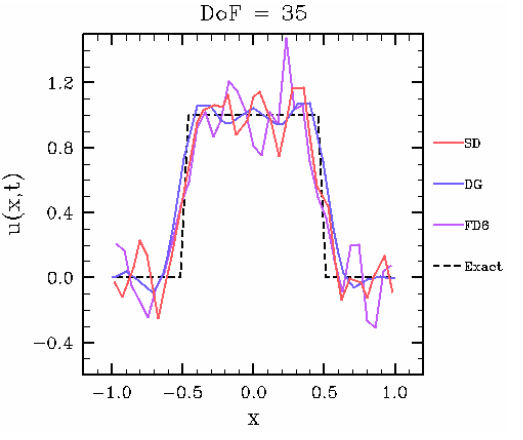


Fig. 4. Plot of nonsmooth (square wave) advection solution with 6th order FD, DG, and SD-LF (35 degrees of freedom)

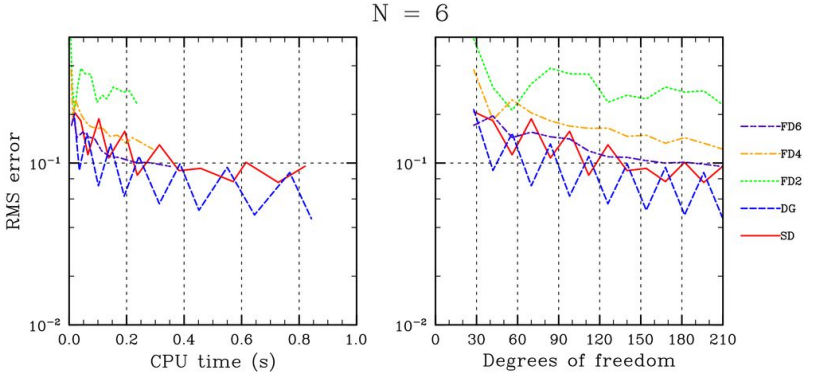


Fig. 5. RMSE in wave equation square wave solution against CPU time (left) and total number of degrees of freedom (right) for 2nd, 4th and 6th order FD, 6th order DG and 6th order SD-LF

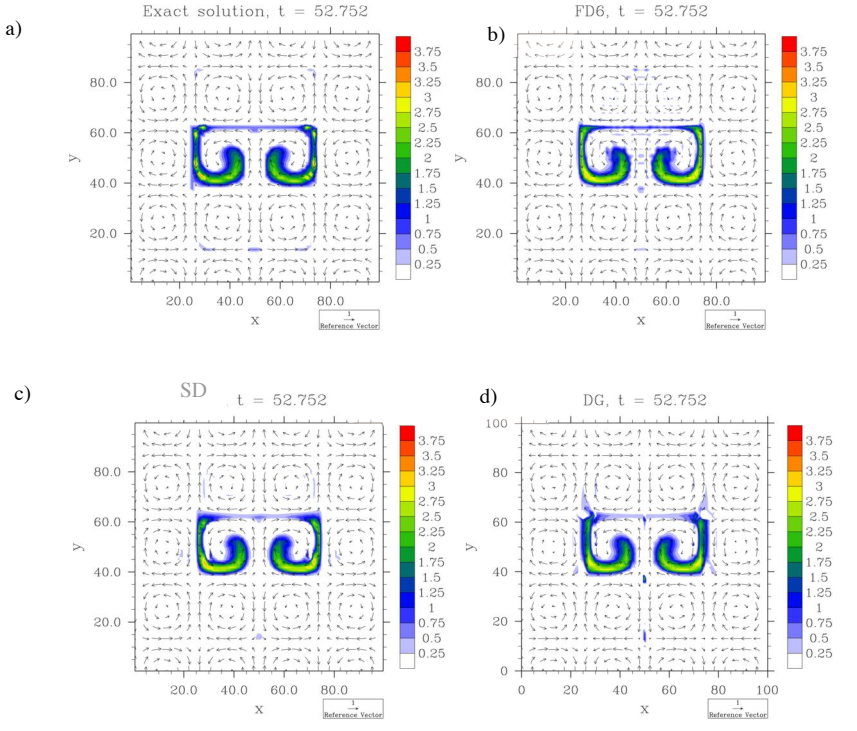


Fig. 6. Smolarkiewicz deformational flow: a) exact solution at time $t_b/5$, and b)-d) numerical solution obtained using 6th order FD, SD-LF and DG respectively (70x70 degrees of freedom) at time $t_b/5$

4 Conclusion

The tests presented show that no one method outperforms the others in a consistent manner. Additional tests in [9] confirm this conclusion. The tests do elucidate the behavior of the different methods however, on different features of advective flows. These are important to keep in mind, especially when resolving at the limit of acceptable resolution.

Acknowledgments. This work has been supported by the US National Science Foundation under CMG grant 0530820. Fruitful discussions with Amik St-Cyr have influenced this work.

References

1. Boyd, J.P.: Chebyshev and Fourier Spectral Methods, 2nd edn. Dover Publications, Mineola (2001)
2. `http://www.cgd.ucar.edu/gds/wanghj/taylor/doe/seam.html` (accessed February 4, 2009)

3. http://www.homme.ucar.edu/ (accessed February 4, 2009)
4. Dennis, J.M., Nair, R.D., Tufo, H.M., Levy, M., Voran, T.: Development of a Scalable Global Discontinuous Galerkin Atmospheric Model. Int. J. Comp. Sci. Eng. (2008) (to appear), http://www.cisl.ucar.edu/css/staff/rnair/ijcse.pdf (accessed February 4, 2009)
5. Kopriva, D.A., Kolias, J.H.: A conservative staggered-grid Chebyshev multidomain method for compressible flows. J. Comp. Phys. 125(1), 244–261 (1996)
6. Liu, Y., Vinokur, M., Wang, Z.J.: Spectral difference method for unstructured grids I: basic formulation. J. Comput. Phys. 216, 780–801 (2006)
7. Van den Abeele, K., Lacor, C., Wang, Z.J.: On the stability and accuracy of the spectral finite difference method. J. Sci. Comp. 37(2), 162–188 (2008)
8. Durran, D.R.: Numerical Methods for Wave Equations in Geophysical Fluid Dynamics, Texts in Applied Mathematic, vol. 32. Springer, Heidelberg (1998)
9. Williams, D.: A Comparison Between Finite Difference and Element Based Spectral Methods for Transport in Meteorological Models, M.S. Thesis, University of Oklahoma (2008)
10. Shchepetkin, A.F., McWilliams, J.C.: Quasi-monotone advection schemes based on explicit locally adaptive dissipation. Mon. Wea. Rev. 126, 1541–1580 (1998)
11. Cockburn, B., Shu, C.-W.: Runge–Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems. J. Sci. Comp. 16(3), 173–261 (2001)
12. Smolarkiewicz, P.K.: The multi-dimensional Crowley advection scheme. Mon. Wea. Rev. 113, 1050–1065 (1982)
13. Staniforth, A., Côté, J., Pudykiewicz, J.: Comments on Smolarkiewicz's deformational flow. Mon. Wea. Rev. 115, 894–900 (1987)

A Non-oscillatory Advection Operator for the Compatible Spectral Element Method

M.A. Taylor[1,⋆], A. St.Cyr[2,⋆⋆], and A. Fournier[2]

[1] Sandia National Laboratories, Albuquerque NM 87185
mataylo@sandia.gov
[2] National Center for Atmospheric Research⋆⋆⋆, Boulder CO 80303

Abstract. The spectral element method is well known as an efficient way to obtain high-order numerical solutions on unstructured finite element grids. However, the oscillatory nature of the method's advection operator makes it unsuitable for many applications. One popular way to address this problem is with high-order discontinuous-Galerkin methods. In this work, an alternative solution which fits within the continuous Galerkin formulation of the spectral element method is proposed. Making use of a compatible formulation of spectral elements, a natural way to implement conservative non-oscillatory reconstructions for spectral element advection is shown. The reconstructions are local to the element and thus preserve the parallel efficiency of the method. Numerical results from a low-order quasi-monotone reconstruction and a higher-order sign-preserving reconstruction are presented.

1 Introduction

The spectral element method (SEM) with inexact numerical integration is a generalized continuous Galerkin method [1]. It is *h-p* capable, relies on globally continuous polynomial basis functions and the equations of interest are solved in integral form. The unique feature of the spectral element method is that if the elements are restricted to quadrilaterals, the integrals can be approximated by highly accurate Gauss-Lobatto quadrature rules within each element. This allows the construction of compactly supported, globally continuous basis and test functions which are orthogonal, leading to a diagonal mass matrix. The diagonal mass matrix allows time-dependent geophysical problems to be solved with simple explicit or semi-implicit methods and thus the method remains efficient while retaining the geometric flexibility of unstructured finite element grids. The method has proven accurate and effective for a wide variety of geophysical problems, including global atmospheric circulation modeling[2,3,4,5,6,7], ocean modeling [8,9], and planetary-scale seismology [10]. The method has unsurpassed

⋆ Supported in part by DOE/BER FWP06-13194.
⋆⋆ Supported in part by DOE DE-FG02-07ER64464 and NSF CMG-0530845.
⋆⋆⋆ NCAR is operated by the University Corporation for Atmospheric Research and sponsored by the National Science Foundation.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 273–282, 2009.

parallel performance. It was used for earthquake modeling by the 2003 Gordon Bell Best Performance winner [11] and has successfully scaled to $\sim 100,000$ processors [12,13].

One caveat of the SEM has its source in the advection operator. For advection, the SEM can achieve excellent accuracy in the L_2 norm mainly because it uses relatively high-degree polynomials (typically between degree 4 and 10). However, the fact that the basis functions are globally continuous makes it difficult to preserve discrete analogs of other important practical properties of advection such as monotonicity and positivity. Traditional SEM results are quite oscillatory [14]. In this work, it is shown how to incorporate local element reconstructions within a strong-stability-preserving (SSP) time-integrator for the compatible SEM formulation, which yield efficient non-oscillatory advection schemes.

1.1 The Spectral Element Method

Let Ω represent our computational domain. We first mesh Ω using a quadrilateral finite-element mesh with M elements denoted $\{\Omega_m\}_{m=1}^{M}$. Here the focus is on the case where Ω is the surface of the sphere, and we employ a cubed-sphere based tiling of the sphere with quadrilaterals as shown in Fig. 1. It is assumed that the mesh has no hanging nodes, and that each element can be C^1 mapped to the reference element $[-1,1]^2$. We denote this map and its inverse by $\boldsymbol{r} = \boldsymbol{r}(\boldsymbol{x};m)$ and $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{r};m)$, where $\boldsymbol{x} = (x^1,x^2)$ are the coordinates of a point in the reference element $[-1,1]^2$ and $\boldsymbol{r} = (r^1,r^2) \in \Omega$. Within $[-1,1]^2$ we work in the space of polynomials up to degree d, denoted

$$\mathcal{P}_d = \operatorname*{span}_{i,j=0}^{d}\{\phi_{ij}\},$$

where $\phi_{ij}(\boldsymbol{x}) = \varphi_i(x^1)\varphi_j(x^2)$ are the cardinal-functions (Lagrange interpolating polynomials) of the degree d Gauss-Lobatto nodes $\xi_i, i = 0,\ldots,d$. The cardinal-function expansion coefficients of a function g are its Gauss-Lobatto node values, so we have

$$g(\boldsymbol{x}) = \sum_{i,j=0}^{d} g(\xi_i,\xi_j)\phi_{ij}(\boldsymbol{x}) \quad \forall g \in \mathcal{P}_d.$$

The SEM uses global piecewise polynomial spaces $\mathcal{H}_d^0$ and $\mathcal{H}_d^1$ defined as

$$\mathcal{H}_d^0 = \{f \in L^2(\Omega) : f(\boldsymbol{r}(\boldsymbol{x};m)) \in \mathcal{P}_d, \forall m\}, \tag{1}$$

$$\mathcal{H}_d^1 = C^0(\Omega) \cap \mathcal{H}_d^0. \tag{2}$$

Functions in $\mathcal{H}_d^0$ are polynomial in the mapped variable within each element, and $\mathcal{H}_d^1$ is the subset of these functions which are continuous across element boundaries. Let $M_d = \dim \mathcal{H}_d^0 = (d+1)^2 M$, and $L = \dim \mathcal{H}_d^1 < M_d$.

For functions $f \in \mathcal{H}_d^0$, we will rely on the cardinal-function expansion local to each element

$$f(\boldsymbol{r}(\boldsymbol{x};m)) = \sum_{i,j=0}^{d} \hat{f}_{ij}^m \ \phi_{ij}(\boldsymbol{x}), \tag{3}$$

Fig. 1. Tiling the surface of the sphere with quadrilaterals. An inscribed cube is projected to the surface of the sphere. The faces of the cubed-sphere are further subdivided to form a quadrilateral grid of the desired resolution. The Gnomonic equal angle projection is used, resulting in a quasi-uniform but non-orthogonal grid [15].

where the expansion coefficients are the function values at the Gauss-Lobatto nodes, $\hat{f}_{ij}^m = f(\boldsymbol{r}(\xi_i, \xi_j; m))$. Since functions in $\mathcal{H}_d^0$ can be multi-valued at Gauss-Lobatto points shared by more than one element, this local expansion representation will contain all such values. For $f \in \mathcal{H}_d^1$, the values at any multiply represented points must all be the same. Note that since $f(\boldsymbol{r}(\boldsymbol{x}; m))$ is a polynomial of degree d in $\boldsymbol{x}$ and there are $d+1$ Gauss-Lobatto points along each edge, then agreement at these points means we also have agreement along the entire edge, as required for $\mathcal{H}_d^1$. We note that a global piecewise cardinal function basis for $\mathcal{H}_d^1$ can be constructed by piecing together appropriate combinations of the ϕ_{ij} for either conforming [4] or non-conforming element meshes [16].

1.2 The SEM Divergence Operator in Curvilinear Coordinates

We denote the 3×3 Jacobian matrix of the mapping from $[-1, 1]^2$ to Ω_m by $\mathbf{J}^m$, with coefficients $(\mathbf{J}^m)_{\alpha\beta} = \partial r^\alpha / \partial x^\beta$ and determinant $J^m = |\mathbf{J}^m|$. A vector $\boldsymbol{v}$ has contravariant components $v^\alpha = \boldsymbol{v} \cdot \nabla x^\alpha$ and covariant components $v_\beta = \boldsymbol{v} \cdot \partial \boldsymbol{r} / \partial x^\beta$. The divergence operator in Ω_m is given by

$$\nabla \cdot \boldsymbol{v} = \frac{1}{J^m} \sum_\alpha \frac{\partial}{\partial x^\alpha} (J^m v^\alpha). \tag{4}$$

To compute this operator, the term $J^m v^\alpha$ that appears is first projected into $\mathcal{H}_d^0$ via interpolation at the Gauss-Lobatto grid points and then this interpolant is differentiated exactly with respect to x^α by differentiating (3). We denote the interpolation operator by $\mathcal{I}$. The sum of partial derivatives are then divided by J^m at the Gauss-Lobatto nodal values and thus

$$\nabla \cdot \boldsymbol{v} \approx \nabla_h \cdot \boldsymbol{v} = \mathcal{I} \left(\frac{1}{J}^m \sum_\alpha \frac{\partial}{\partial x^\alpha} (\mathcal{I}(J^m v^\alpha)) \right) \in \mathcal{H}_d^0, \tag{5}$$

where $\nabla_h \cdot ()$ is the SEM divergence operator. In what follows, only the SEM operators will be employed, never the continuum operators, and thus the h subscript will be dropped.

1.3 The SEM Inner Product

Instead of using exact integration of the basis functions as in finite-element method, the SEM uses a Gauss-Lobatto quadrature approximation for the inner product. The following unlabeled integral is defined as the usual area weighted integral over the entire domain Ω. This integral is written as a sum of integrals over the set $\{\Omega_m\}$ of elements used to decompose the domain:

$$\int fg = \sum_{m=1}^{M} \int_{\Omega_m} fg = \sum_{m=1}^{M} \iint_{[-1,1]^2} f\,|_{\Omega_m} g\,|_{\Omega_m} J^m \,\mathrm{d}x^1\,\mathrm{d}x^2. \tag{6}$$

The integral over $[-1,1]^2$ is approximated as

$$\langle f,g\rangle_{\Omega_m} = \sum_{i,j=0}^{d} w_i w_j J^m(\xi_i,\xi_j)\hat{f}^m_{ij}\hat{g}^m_{ij} \tag{7}$$

by using the Gauss-Lobatto quadrature points $\{\xi_k\}_{k=0}^d$ and weights $\{w_k\}_{k=0}^d$. The SEM approximation to the global integral is then naturally defined as

$$\langle f,g\rangle = \sum_{m=1}^{M} \langle f,g\rangle_{\Omega_m} \simeq \int fg$$

which is extended to vectors in the usual manner,

$$\langle \boldsymbol{u},\boldsymbol{v}\rangle_{\Omega_m} = \sum_{i,j=0}^{d} w_i w_j J^m(\xi_i,\xi_j)\boldsymbol{u}(\boldsymbol{r}(\xi_i,\xi_j;m))\cdot\boldsymbol{v}(\boldsymbol{r}(\xi_i,\xi_j;m)).$$

1.4 The Discrete Divergence Theorem within an Element

We will also define a Gauss-Lobatto quadrature approximation to the line integral over the boundary of Ω_m as

$$\langle \boldsymbol{v}\cdot\hat{n}\rangle_{\partial\Omega_m} \simeq \oint_{\partial\Omega_m} \phi\boldsymbol{v}\cdot\hat{n}\,ds,$$

with ds being the arc length measure and $\hat{n}$ the outward unit normal. Some algebra will show that the natural Gauss-Lobatto approximation to this integral in curvilinear coordinates is

$$\langle \boldsymbol{v}\cdot\hat{n}\rangle_{\partial\Omega_m} = \sum_{\alpha\neq\beta}\sum_{i=0}^{d} w_i (J^m\boldsymbol{v}\cdot\nabla x^\beta)\Big|_{x^\alpha=\xi_i, x^\beta=-1}^{x^\alpha=\xi_i, x^\beta=1} \tag{8}$$

where the inner sum is over all Gauss-Lobatto points along an element edge and the outer sum is over $(\alpha,\beta)=(1,2)$, representing the element edges where $x^2=\pm1$ and $(\alpha,\beta)=(2,1)$, representing the element edges where $x^1=\pm1$. Note that the corner nodes, which are shared by two edges of Ω_m, appear twice in this sum.

The compatible SEM employs discrete analogs of several important integral properties of the divergence, gradient and curl operators [17].The key property

we need here is the divergence theorem within an element. In the particular, the SEM discrete analog of

$$\int_{\Omega_m} \nabla \cdot \boldsymbol{v} = \oint_{\partial\Omega_m} \boldsymbol{v} \cdot \hat{n}\, \mathrm{d}s \tag{9}$$

is given by

$$\langle 1, \nabla \cdot \boldsymbol{v} \rangle_{\Omega_m} = \langle \boldsymbol{v} \cdot \hat{n} \rangle_{\partial\Omega_m}, \quad \forall \boldsymbol{v} \in \boldsymbol{\mathcal{H}}_d^1. \tag{10}$$

This property shows that the SEM will be locally conservative when solving equations in conservation form. In the SEM one does not need to compute the flux term, but Eq. 10 shows that the equation has a flux formulation and thus is locally conservative with respect to the element mass $\langle 1, \cdot \rangle_{\Omega_m}$.

2 The SEM Locally Conservative Advection Operator

Consider the advection operator on the surface of the sphere,

$$\frac{\partial h}{\partial t} = -\nabla \cdot h\boldsymbol{v},$$

with $\boldsymbol{v}$ prescribed and $\nabla \cdot \boldsymbol{v} = 0$. This problem is analyzed using a forward Euler time step, and thus the results will naturally extend to higher order SSP time-stepping methods which are convex combinations of forward Euler steps. The resulting SEM discretization finds $h(t + \Delta t) \in \mathcal{H}_d^1$ such that

$$\langle \psi, h(t + \Delta t) \rangle = \langle \psi, h(t) \rangle - \Delta t \, \langle \psi, \nabla \cdot h(t)\boldsymbol{v}(t) \rangle \qquad \forall \psi \in \mathcal{H}_d^1.$$

The latter is equivalent to the following two-step process:

1. Advance the solution locally within each element,

$$h^* = h(t) - \Delta t \nabla \cdot h(t)\boldsymbol{v}(t). \tag{11}$$

2. Let $h(t+\Delta t)$ be the projection of h^* into $\mathcal{H}_d^1$. The projection is given by the unique $h(t + \Delta t) \in \mathcal{H}_d^1$ such that

$$\langle \psi, h(t + \Delta t) \rangle = \langle \psi, h^* \rangle \qquad \forall \psi \in \mathcal{H}_d^1. \tag{12}$$

Step one computes an $h^* \in \mathcal{H}_d^0$ that in general will not be globally continuous and thus not in $\mathcal{H}_d^1$. Projecting h^* into $\mathcal{H}_d^1$ in the second step requires inverting the SEM mass matrix.

3 A Quasi-Monotone SEM Advection Operator

It is first shown that a low-order quasi-monotone SEM advection scheme can be obtained by introducing a reconstruction step between the two steps of the algorithm given above:

1. Advance the solution locally within each element using Eq. 11.
2. Let h^{**} be the the result of a mass-preserving bilinear reconstruction of h^* with slope limited so that no new extrema are created.
3. Let $h(t+\Delta t)$ be the projection of h^{**} into $\mathcal{H}_d^1$.

To show that this method is quasi-monotone, we first establish

Theorem 1. *Suppose Δt is chosen such that*

$$\Delta t \left| \langle h(t)\boldsymbol{v}(t)\cdot\hat{n}\rangle_{\partial\Omega_m} \right| \leq \langle 1, |h(t)|\rangle_{\Omega_m} \qquad \forall m$$

then step 1 (the SEM local element update) obeys a monotone-element-mean property,

$$\min_{i,j} h(\boldsymbol{r}(\xi_i,\xi_j;m),t) \leq \langle 1, h^*\rangle_{\Omega_m} / \langle 1,1\rangle_{\Omega_m} \leq \max_{i,j} h(\boldsymbol{r}(\xi_i,\xi_j;m),t).$$

Proof. For $\langle 1, |h(t)|\rangle_{\Omega_m} \neq 0$, such a Δt can always be chosen. Otherwise $h=0$ and the inequality is satisfied for all Δt. Note that the restriction on Δt is a standard CFL condition, since $\langle 1,1\rangle_{\Omega_m} / \langle 1\rangle_{\partial\Omega_m}$ is proportional to element edge length.

To show that step 1 has the monotone-element-mean property, we first show that if $h(t)\geq 0$, then $\langle 1, h^*\rangle_{\Omega_m} \geq 0$. By Eq. 10 and the fact that $\langle 1, |h(t)|\rangle_{\Omega_m} = \langle 1, h(t)\rangle_{\Omega_m}$, we have

$$\langle 1, h^*\rangle_{\Omega_m} = \langle 1, h(t)\rangle_{\Omega_m} - \Delta t \langle 1, \nabla\cdot h(t)\boldsymbol{v}(t)\rangle_{\Omega_m} \tag{13}$$

$$= \langle 1, h(t)\rangle_{\Omega_m} - \Delta t \langle h(t)\boldsymbol{v}(t)\cdot\hat{n}\rangle_{\partial\Omega_m} \geq 0. \tag{14}$$

Now consider

$$g_1(t) = h(t) - \min_{i,j} h_t(\boldsymbol{r}(\xi_i,\xi_j;m),t) \geq 0 \tag{15}$$

$$g_2(t) = \max_{i,j} h(\boldsymbol{r}(\xi_i,\xi_j;m),t) - h(t) \geq 0. \tag{16}$$

Applying the SEM advection step 1 to both $g_1(t)$ and $g_2(t)$, we have that $(1, g_1^*)_{\Omega_m} \geq 0$ and $(1, g_2^*)_{\Omega_m} \geq 0$, which is equivalent to the monotone-element-mean property.

Since h^* computed in step 1 will not contain any new extrema relative to the min and max within Ω_m, the slope limited reconstruction h^{**} computed in step 2 will obey the same property. It will be non-oscillatory, but only quasi-monotone since for high polynomial degree, if $h(t)$ is highly oscillatory within the element, it is possible that h^{**} will contain local extrema in one region of the element even though it does not contain any new extrema with respect to the element min and max values. The third and final step, applying the SEM projection operator, is a Jacobian weighted averaging of the values computed at different elements for the shared edges and corner points, and is thus monotone preserving.

This quasi-monotone scheme is far from optimal. Numerical results suggest it is only 2^{nd}-order accurate when applied to smooth problems. It remains an open

problem to determine if higher-order monotone reconstructions exist and to determine if an an exactly monotone reconstruction exists. Our initial results at higher-order reconstructions have focused on retaining only the sign-preserving property. We note that a conservative sign-preserving reconstruction always exists since in the worse case the reconstruction can simply set h^{**} to the element average which is conservative and always positive by Theorem 1.

4 Numerical Results

We now compare the three spectral element advection schemes: no reconstruction, the quasi-monotone reconstruction described above and a sign-preserving reconstruction. We have implemented these methods into the National Center for

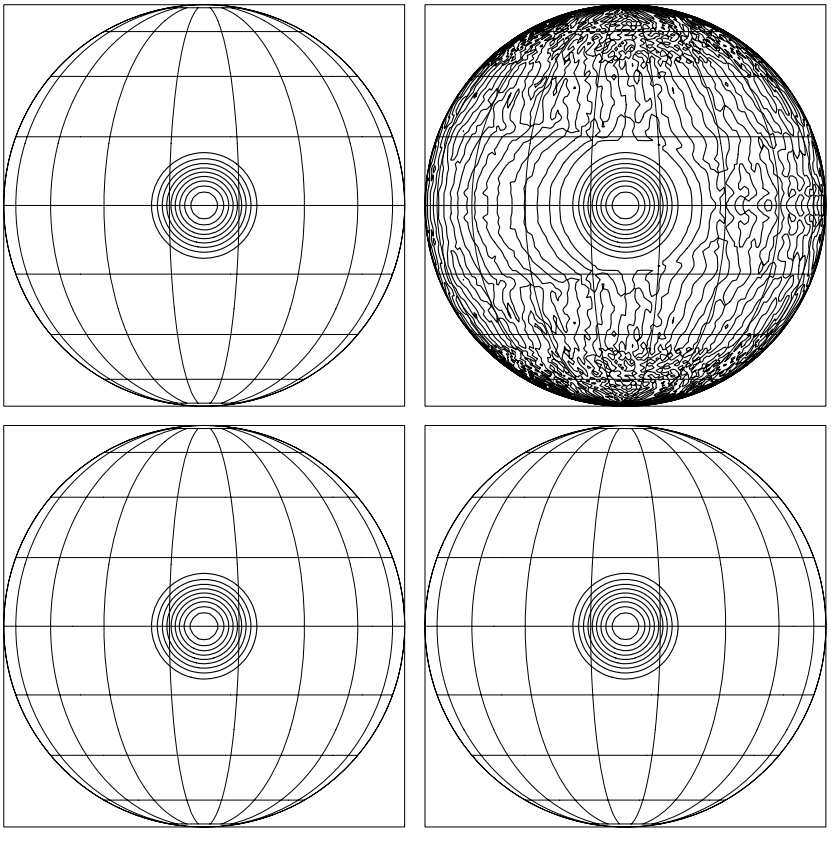


Fig. 2. Contour plots of the cosine-bell test case. Shown are the initial condition (upper left) and the solution from the un-limited advection scheme (upper right), the quasi-monotone scheme (lower left) and the sign-preserving scheme (lower right). Contour lines are drawn for $h = 0$ to $h = 1000$ with an increment of 100.

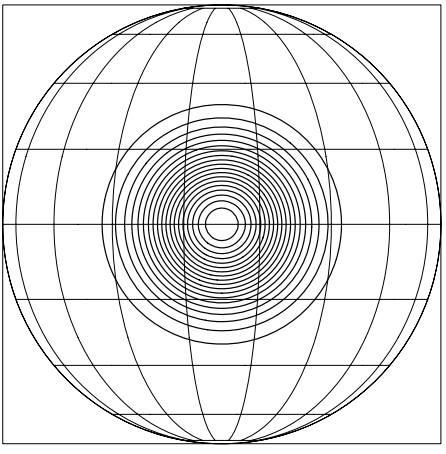


Fig. 3. Contour plot of the initial condition for the Gaussian test case. Contour lines are drawn for $h = 0$ to $h = 1000$ with an increment of 50.

Atmospheric Research's High Order Method Modeling Environment (HOMME) [5]. Within each face of the cubed sphere, we use an $ne \times ne$ grid of elements. Within each element we use degree $d = 3$ polynomial basis functions, which is formally 4^{th}-order accurate. The average grid spacing at the equator is $360/(12\,ne)$ degrees. We do not describe in detail our initial reconstruction algorithms since they are presented here only to demonstrate the potential of our approach.

We start with the pure advection test from the well known suite of shallow-water test cases on the sphere[18]. The latter concerns the advection around the sphere of a cosine bell with compact support. The velocity is fixed (rigid rotation about the north-south axis) and the equation is integrated for 12 days or one full rotation around the sphere. Contour plots of the initial condition $h(0)$ and results from the 3 advection schemes after 12 days are shown in Fig. 2. As expected, the non-limited advection scheme is quite oscillatory, especially in the region where the solution should be zero, but has very little dissipation: the maximum of h is reduced from 1000.0 to 994.0 after 12 days, while the minimum is -5.97. Both the sign-preserving and monotone schemes completely eliminate these oscillations but have slightly more dissipation, reducing the maximum after 12 days to 992.7 and 959.0, respectively. The minimum of h is zero for both of these methods.

The cosine bell has a kink at the edge of the bell where $h = 0$ and thus none of the methods can achieve convergence greater than second order. To study the order of accuracy of these methods, we modify the test and instead advect a smooth Gaussian hill, shown in Fig. 3. We compute l_2 and l_∞ errors using the same normalization as used for the first test as specified in [18]. These errors are plotted in Fig. 4 for resolutions of $ne = 9$ up to $ne = 41$. As expected, the non-limited spectral element advection is 4^{th}-order accurate even on the non-orthogonal unstructured cubed-sphere grid. The sign-preserving advection

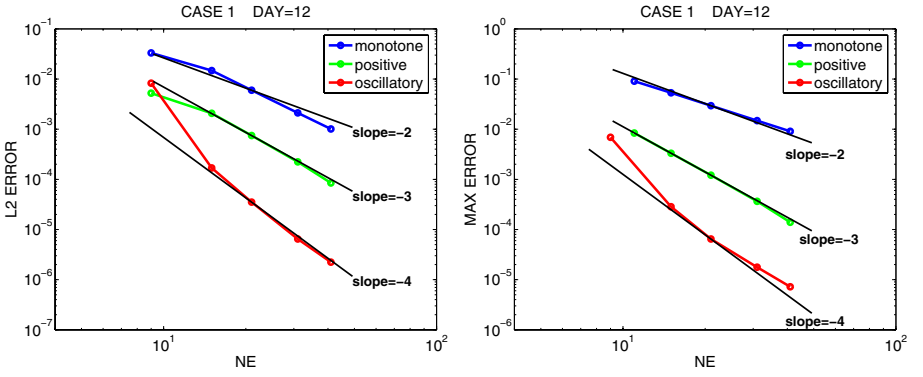


Fig. 4. Convergence of the l_2 and l_∞ error as a function of resolution parameter ne. The non-limited spectral element scheme is labeled oscillatory, while positive refers to the scheme with a positive-preserving reconstruction and monotone is the scheme with a quasi-monotone reconstruction.

scheme loses one order of accuracy, and the monotone scheme reduces the accuracy of the scheme to second order.

5 Conclusions

In this work it was shown how to naturally include a function-limiting procedure within the compatible SEM formulation for the time-dependent pure advection equation. We then demonstrated that quasi-montone and sign-preserving advection schemes are obtainable by the SEM. The schemes are far from optimal and thus future work will focus on fully monotone and higher-polynomial-degree reconstruction procedures.

References

1. Maday, Y., Patera, A.T.: Spectral element methods for the incompressible Navier Stokes equations. In: Noor, A.K., Oden, J.T. (eds.) State of the Art Surveys on Computational Mechanics, pp. 71–143. ASME, New York (1987)
2. Taylor, M., Tribbia, J., Iskandarani, M.: The spectral element method for the shallow water equations on the sphere. J. Comput. Phys. 130, 92–108 (1997)
3. Giraldo, F.X.: A spectral element shallow water model on spherical geodesic grids. International Journal for Numerical Methods in Fluids 35, 869–901 (2001)
4. Fournier, A., Taylor, M., Tribbia, J.: The spectral element atmosphere model (SEAM): High-resolution parallel computation and localized resolution of regional dynamics. Mon. Wea. Rev. 132, 726–748 (2004)
5. Thomas, S., Loft, R.: The NCAR spectral element climate dynamical core: Semi-implicit eulerian formulation. J. Sci. Comput. 25, 307–322 (2005)
6. Dennis, J., Fournier, A., Spotz, W.F., St -Cyr, A., Taylor, M.A., Thomas, S.J., Tufo, H.: High resolution mesh convergence properties and parallel efficiency of a spectral element atmospheric dynamical core. Int. J. High Perf. Comput. Appl. 19, 225–235 (2005)

7. Wang, H., Tribbia, J.J., Baer, F., Fournier, A., Taylor, M.A.: A spectral element version of CAM2. Monthly Weather Review 135 (2007)
8. Haidvogel, D., Curchitser, E.N., Iskandarani, M., Hughes, R., Taylor, M.A.: Global modeling of the ocean and atmosphere using the spectral element method. Atmosphere-Ocean Special 35, 505–531 (1997)
9. Molcard, A., Pinardi, N., Iskandarani, M., Haidvogel, D.: Wind driven circulation of the mediterranean sea simulated with a spectral element ocean model. Dynamics of Atmospheres and Oceans 35, 97–130 (2002)
10. Komatitsch, D., Tromp, J.: Spectral-element simulations of global seismic wave propagation - I. validation. Geophys. J. Int. 149, 390–412 (2002)
11. Komatitsch, D., Tsuboi, S., Ji, C., Tromp, J.: A 14.6 billion degrees of freedom, 5 teraflops, 2.5 terabyte earthquake simulation on the earth simulator. In: Proceedings of the ACM / IEEE Supercomputing SC 2003 conference (2003)
12. Bhanot, G., Dennis, J.M., Edwards, J., Grabowski, W., Gupta, M., Jordan, K., Loft, R.D., Sexton, J., St-Cyr, A., Thomas, S.J., Tufo, H.M., Voran, T., Walkup, R., Wyszogrodski, A.A.: Early experiences with the 360TF IBM BlueGene/L platform. International Journal of Computational Methods 5, 237–253 (2008)
13. Taylor, M.A., Edwards, J., St-Cyr, A.: Petascale atmospheric models for the community climate system model: New developments and evaluation of scalable dynamical cores. J. Phys. Conf. Ser. 125(012023) (2008)
14. Iskandarani, M., Levin, J., Choi, B.J., Haidvogel, D.: Comparison of advection schemes for high-order hp finite element and finite volume methods. Ocean Modelling 10, 233–252 (2005)
15. Rančić, M., Purser, R., Mesinger, F.: A global shallow-water model using an expanded spherical cube: Gnomonic versus conformal coordinates. Q. J. R. Meteorol. Soc. 122, 959–982 (1996)
16. Fournier, A., Rosenberg, D., Pouquet, A.: Dynamically adaptive spectral-element simulations of 2d incompressible navier-stokes vortex decays. Geophysical and Astrophysical Fluid Dynamics (2009) (to appear)
17. Taylor, M.A., Edwards, J., Thomas, S., Nair, R.: A mass and energy conserving spectral element atmospheric dynamical core on the cubed-sphere grid. J. Phys. Conf. Ser. 78(012074) (2007)
18. Williamson, D.L., Drake, J.B., Hack, J.J., Jakob, R., Swarztrauber, P.N.: A standard test set for numerical approximations to the shallow water equations in spherical geometry. J. Comput. Phys. 102, 211–224 (1992)

Simulating Particulate Organic Advection along Bottom Slopes to Improve Simulation of Estuarine Hypoxia and Anoxia

Ping Wang[1] and Lewis C. Linker[2]

[1] University of Maryland Center for Environmental Science, Chesapeake Bay Program, 410 Severn Avenue, Annapolis, MD 21403, USA
pwang@chesapeakebay.net
[2] US Environmental Protection Agency/CBPO, 410 Severn Ave., Suite 109, Annapolis, MD 21403, USA
linker.lewis@epa.gov

Abstract. In a coupled hydrodynamic and water quality model, the hydrodynamic model provides forces for movement of simulated particles in the water quality model. A proper simulation of organic solid movement from shallow to deep waters is important to simulate summer hypoxia in the deepwater. It is necessary to have a full blown particle transport model that focuses organic particulates' resuspension and transport. This paper presents an approach to move volatile solids from the shoals to the channel by simulating movement of particulate organics due to slopes based on an example in the Chesapeake Bay eutrophication model. Implementations for the simulation of this behavior in computer parallel processing are discussed.

Keywords: hydrodynamic model, estuarine model, movement along slope, particulate organic transport, parallel processing.

1 Introduction

The Chesapeake Bay Estuarine (Water Quality and Sediment Transport) Model is designed to simulate the current estuarine eutrophication in the Chesapeake and to examine nutrient and sediment reductions to restore water quality [1]. It simulates algal blooms due to excessive nutrient inputs, and the subsequent decay of organics and reduction in dissolved oxygen in deep water, particularly in the middle channel of the main-stem Bay and tidal tributaries in the summer. Volatile, or reactive organics in the Bay are in dissolved and particulate forms and the particulate organics are also referred to as volatile solids or volatile suspended solids. We propose that volatile solids deposition to the channel bed come from other areas of the estuary, including the near-shore shallow waters, other then solely algal production and settling in the water column overlying the deep channel [2, 3]. This is based on the observation that a simple mass balance of the algal production over the deep channel of the mainstem Chesapeake is insufficient for generation of the observed anoxia in the deep water.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 283–292, 2009.

The Water Quality and Sediment Transport Model is coupled with the Estuarine Hydrodynamic Model [4] which provides forcing for particle transport in water columns. The hydrodynamic model also provides bottom shears stress for the simulation of scour and resuspension of sediment from bed. Besides movement due to hydrodynamic forces, sediment can also sink down and move along bed slope due to gravity [5]. The Chesapeake Bay Estuarine Model assigned settling velocities to simulate sinking from upper model cells toward sediment beds for different sediment classes, however without considering movement along bed slope [6]. The resuspension due to bottom shear is under development for inorganic solids, while the resuspension for organic solids are only partially simulated by reducing net settling rates based on model calibration. In this context, once organic solids settle on bed, they will no-longer be scoured or resuspended. This causes insufficient transport of organic solids from shallow to deep waters and causes insufficient oxygen demand to simulate the observed anoxia and hypoxia. A remedy to this in the earlier phase of model, i.e., with a grid of 13,000 cells [6], was by adjusting some parameters through model calibration that yielded reasonable simulation of dissolved oxygen (DO) as shown in Figure 1. The circle symbols are observed DO and the dots are simulated DO in the 13,000-cell model. However, as the model grid was refined by an order of magnitude to the current 57,000 cells many shallow water and shoal cells became less connected with deep water and the adjusted parameters in the old model calibration became less optimal, as pointed out by Michael Kemp of the University of Maryland Center for Environmental Science (personal communication) and illustrated by Figure 2. Here we assume that the coarse grid has two cells horizontally, from the right shoal to the channel (Figure 2a); while the finer grid has four cells horizontally (Figure 2b). The materials from the areas of downward arrows can reach the channel bottom cell (letters A and B) more easily for the coarser grid than for the finer grid. This is one of the reasons causing insufficient organic material delivered to deep waters. Even the new refined grid model tried to optimize the same parameters as those in the coarse grid, the calibration of oxygen demand in deep waters is degraded [7]. The plus symbols in Fig. 1 represent the DO simulation in the refined 57,000 cell model.

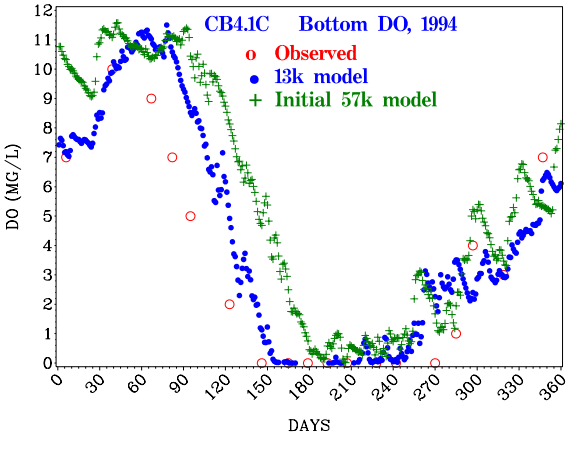


Fig. 1. DO simulations in a coarse grid model (13k grid) and after grid refinement (57k grid)

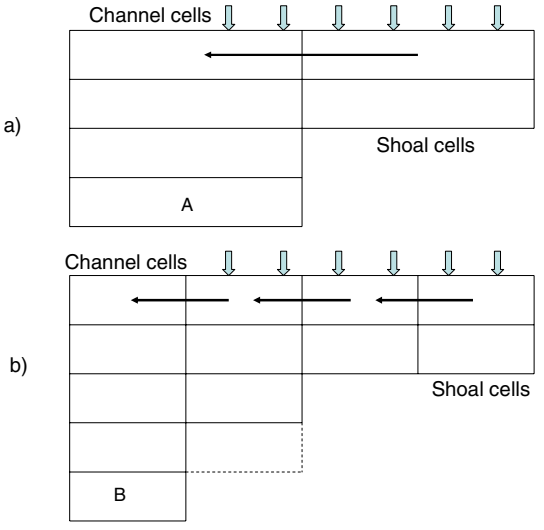


Fig. 2. Schematic coarser (a) and finer (b) grids, showing differences in potential receiving materials from the surface (the downward block-arrows) by channel bed cells (A and B) of the two grids: Cell A of the coarser grid has more chance to receive input materials than Cell B of the finer grid

In general, if a model simulates well on major mechanisms, a refinement of grid would improve simulation. The degradation after grid refinement may be due to the model discounts some important mechanisms which cannot be simulated well by some adjusted parameters used in the initial coarse grid. This paper explores a method to simulate one of the missing mechanisms, i.e., movement of volatile solids along slopes, so that to better simulate volatile solid movement and oxygen demand in channel for the Chesapeake Bay Estuarine Model.

2 Method

2.1 Basic Model

The Chesapeake Bay Estuarine Model, a coupled 3-dimensional finite-difference/finite-volume CH3D Hydrodynamic Model and CE-QUAL-ICM Water Quality Model, is used. The model grid uses the Z-grid structure. In this paper, the implementation of slope movement for model is based on the refined 57k grid.

Daily loads to the model were provided by the Chesapeake Bay Watershed Model. The water quality model simulates major nutrient cycles, including algal growth and decay, involving 36 state variables. Particle movement in the water is controlled by hydrodynamic forces of advection. The CH3D Hydrodynamic Model simulates physical processes impacting estuarine circulation and vertical mixing, that includes tides, freshwater inflows, wind, density effect by salinity and temperature, turbulence, and the Coriolis effect [8]. The basic equations are:

$$\partial u/\partial x + \partial v/\partial y + \partial w/\partial z = 0$$

$$\partial u/\partial t + \partial u^2/\partial x + \partial uv/\partial y + \partial uw/\partial z = f\, v - 1/\rho\, \partial P/\partial x + \partial[A_H\, \partial u/\partial x\,]/\partial x + \partial[A_H\, \partial u/\partial y]/\partial y + \partial[A_V\, \partial u/\partial z\,]/\partial z$$

$$\partial v/\partial t + \partial v^2/\partial y + \partial uv/\partial x + \partial vw/\partial z = -f\, u - 1/\rho\, \partial P/\partial y + \partial[A_H\, \partial v/\partial x\,]/\partial x + \partial[A_H\, \partial v/\partial y\,]/\partial y + \partial[A_V\, \partial v/\partial z\,]/\partial z$$

$$\partial P/\partial z = -\rho\, g$$

where, (u, v, w) = velocities in (x, y, z) directions, t = time, f = Coriolis parameter, ρ=density, P=pressure, A_H = horizontal turbulent, A_V,= vertical turbulent, and g = gravitational acceleration.

For sediment material, S, to transport:

$$\partial S/\partial t + \partial uS//\partial x + \partial vS/\partial y + \partial wS/\partial z = \partial[K_H\, \partial S/\partial x\,]/\partial x + \partial[K_H\, \partial S/\partial y\,]/\partial y + \partial[K_V\, \partial S/\partial z\,]/\partial z$$

where, K_H = eddy coefficient for horizontal turbulent, and K_V = eddy coefficient for vertical turbulent. The vertical turbulence is handled by using the concept of eddy viscosity and diffusivity to represent the velocity and density correlation terms. They are computed from main flow characteristics using a method developed by Donaldson [9] and Sheng [10].

Besides the transport, settling of volatile solids in the water column is calculated:

$$\partial S/\partial t = [\text{transport by hydrodynamic forces}] + S_U\,(W/dz) - S\,(W/dz)$$

where, S_U = solid in a cell above, W = settling velocity in water column, and dz = cell thickness.

Since the resuspension of volatile solid by bottom shears stress is not simulated, a remedy is set for the cells that interface the bed sediment: use net settling velocity, Wnet, which is after a subtraction from the settling velocity (W) to account for the un-simulated resuspension. The Wnet is obtained empirically through model calibration. Thus, the volatile solid in the bottom cell is calculated by:

$$\partial S/\partial t = [\text{transport by hydrodynamic forces}] + S_U\,(W/dz) - S\,(Wnet/dz)$$

The movement of materials in water columns is through the faces connecting model cells. In a Σ-grid, bottom cells are connected with vertical faces. While in a Z-grid, bottom cells are not always connected. There, cells among layers are divided by horizontal faces (Fig. 2), including the bottom face of bottom cells even there is slope on bed (as the bold curves in Figure 3). The grid uses a different number of layers to represent different water depths. Particle transport by hydrodynamic forces from a bottom cell (e.g., A) to another bottom cell (e.g., C) in the adjacent water column where there is a layer difference as in this example, the particle will first go parallel to the adjacent cell of the same layer (e.g., B) through a vertical face, then move downward to the bottom cell through a horizontal face (as by the line-arrows a->b->c).

2.2 Adding Movement along Slopes

The initial Chesapeake Bay Water Quality Model does not simulate movement of volatile organic solids along bottom slopes. We add an additional simulation of particulate organic movement along bottom slopes besides their movement by hydrodynamic forces.

We need to connect bottom cells and determine slope directions. This task is easier to implement in a Σ-grid, since the bottom cells are physically connected.

In a Z-grid the bottom cells are not always physically connected by their faces. The movement along slopes among bottom cells (star symbol in Figure 3) is through direct links, as shown by the block-arrows in Figure 3. In order to implement movement along the bottom slopes in a Z-grid we need to set up computationally an image of the bottom to surface cells. Surface cells are in the same layer and are always connected. The looping computation of solid movement among bottom cells can use the linkages among their counterpart surface cells' image. Nevertheless, for the transport in water columns among bottom cells, materials still need to go the "detour" routes (Fig. 3, line-arrows).

The degrees of slope angles affect the movement along slopes. The angle of slope (α) can be determined from the distance between centers of two adjacent cells (c) and their bathymetry difference (d): $\alpha = \tan^{-1}(d/c)$.

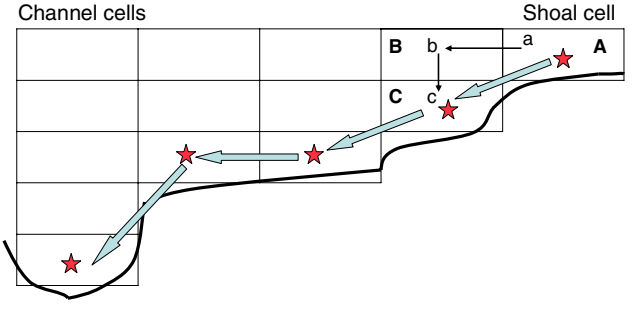


Fig. 3. Schematic graph showing transport simulation among bottom cells in a z-grid. Note: Letters a, b, c with line-arrows are the hydrodynamic movements from cell A to cell C. The movement along slopes among bottom cells (stars) can be implemented through direct links by the block-arrows, which was not simulated by the existing model.

The weight of a solid particle, minus its buoyancy, is the net downward force, $f_1 = m*g - b$, where m is mass of solid, g is gravitational acceleration, b is buoyancy $= (m/\rho s)*\rho w*g, = m*g*\rho w/\rho s$, ρw is density of water, and ρs is the density of the solid. The resultant force along the slope, $f_2 = f_1 * \sin(\alpha)$. The net force along the slope, $f_3 = f_2 - R$, where R is resistance by water and bed. The acceleration of the movement (a) by the net force along the slope is: $a = f_3/(m*g - b)/g = f_3/[m(1 - \rho w/\rho s)]$. The travel distance (s) in a time step (t) if the initial velocity assumes zero: $s = a*t * t /2 = t^2 * f_3/[2m(1 - \rho w/\rho s)]$. The horizontal moving distance in the Z-grid, $h = s * \cos(\alpha)$. The ratio of h / c is proportional to the fraction of the materials moving from shoal cells toward center cells. Thus, the model is able to simulate an additional movement of solid particles along bed slopes.

We compare DO simulation by the 13k calibration, the initial 57k model (Figure 1), and the improved 57k model with movement along slopes (Figure 4, presented later).

2.3 Slope Setup in Grid Computation

Slopes exist among all bed cells. We may simulate movement along slopes for all bottom cells. Alternatively, we may only consider significant slopes, such as across channels, since we mainly want to improve DO simulation in channel and slope is significant across channels. Correspondingly, there are two options to compute movement along slopes among bottom cells: 1) referencing all bottom cells and considering slopes between the reference cell and its adjacent cells; 2) referencing channel bottom cells and considering slopes to each reference cell from its shoal cells on two sides.

In the first option, for each reference cell (Table 1, column 1), there are maximum 4 adjacent cells (columns 2-5; zero cell number means the corresponding adjacent cell does not exist at grid boundary). A reference cell can either receive input from, or provide output to, its adjacent cells, depending on slope toward to or away from the cell. The computation of transport along slope between two cells loops all bottom cells (i.e., the reference cells) for four rounds for their four sides of adjacent cells. To avoid double counting, the transport between two cells is computed only when the reference cell receives input, while the associated adjacent cell reduces the same amount. The disadvantage of this option is that it needs to loop all bottom cells which prolongs computing time.

The second option does not consider all slopes in the model grid. We only reference channel cells (Table 2, Column 2; Column 1 is total shoal cells on one side from the channel cell). Then, we list the corresponding shoal cells, from deep to shallow for each side (Table 2, begin from Column 3). The cells one-by-one receive materials from its shallower adjacent ones. Since there are generally two shoal sides for a channel cell, therefore, a channel cell appears twice in Column 2, while the shoal cells only appear once. Note: if there are two dips in one transact, then they should be separated from the shallowest cell between the dips, and each dip cell is a reference cell.

Table 1. Bottom cell linkages for slope by referencing all bottom cells

Reference cell	Cell (left adjacent)	Cell (right adjacent)	Cell (upper adjacent)	Cell (lower adjacent)
1	101	571	0	2
2	102	572	1	3
..				
..				
102	***	***	***	***
..				
..				
..				
11064	0	10002	11063	0

Note: Numbers for the cells are not actual. 0 indicates no corresponding adjacent cell, due to the adjacent to grid boundary.

Table 2. Bottom cell linkages for slope by referencing channel bottom cells

Total side cells	Reference channel cell	Cells from channel to shoal (on either one side) ----------→				
4	201	410	511	620	721	
2	201	121	98			
5	232	440	541	649	751	862
3	232	151	102	74		
	..					
	..					
	..					
3	999	***	***	***		
5	999	***	***	***	***	***

Note: Numbers for the cells are not actual.

2.4 Implementing Parallel Processing

The Chesapeake Bay Water Quality Model parallel processing used domain decomposition for model cells. The decomposition is based on the numbers of total surface cell, total model cell, and total faces that connect cells, while does not specifically count channel cells and their shoal cells. Note: each bottom cell has a corresponding surface cell, therefore, bottom cells can be referred using surface cells. In this context, looping bottom cells in a decomposed domain can be expressed with looping surface cells.

In the first option of referencing cells in section 2.3, it is relatively easy to decompose the cells in Table 1 into domains, since it deals with all surface cells in the ordinary order. Therefore, it can use the general decomposition routine in the main program. However, because the surrounding 4 cells do not always split into a same domain, each of the four side cells needs to be handled separately to associate with the reference cells.

The second option of referencing cell focuses on channel cells and their shoal cells. This method saves computing time, but additional efforts and cautions are needed in domain decomposition, since the channel (referencing) cells are not specified in the main program's domain decomposition, and the lateral cells in one profile of a reference channel cell may be decomposed into different domains. A reference cell in a decomposed Table 2 may not necessarily be a channel cell, but should be the deepest cell among the cells in the decomposed profile in that domain.

3 Results and Discussion

3.1 Comparing DO Simulations by Slope Movement and the Initial Model

The plus symbols in Figure 4 represent the simulation of DO by the initial 57k grid model calibration, and the stars are the DO simulation after the implementation of the simulation of slope movement. The simulation of volatile solids' movement along the

bottom slope yields lower DO (the star symbols) than the simulation by the initial 57K grid model (the plus symbols), and closer to the observed (the circle symbols). Consistently, the amounts of volatile solids in channel cells are higher in the revised 57k model than the initial 57k model. This supports the idea that the inability in the simulation of anoxia in the channel may be due in part to the lack of transport of volatile suspended sediments from the shallows to the channel, and that implementing movement of particulate organics along slopes can improve the simulation. In models of this type without an explicit simulation of resuspension and transport of particulate organics this approach may be important in order to represent the role of primary production and its fate [11].

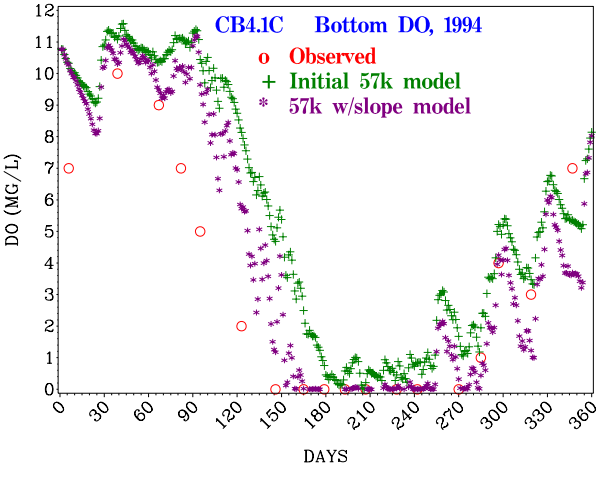


Fig. 4. DO simulation in 57k grid, before and after the simulation of slope movement

3.2 Significance of Slope Movements along Axial versus Lateral Directions

In option 2 of selecting cells we only reference channel bed cells and calculate the movement toward channel from the shoal in the lateral direction, but not the slope movement along the axial direction. This is based on the following two facts. A) The Bay's bathymetry shows that slope gradients are less significant along the axial direction than the lateral direction. B) The hourly hydrodynamic flow fields in the main stem bay and its tributaries indicate that the flow vectors along the channel axial direction are dominant, about 3 to 5 times the lateral direction, except during full high or low tide. Mass fluxes by currents are stronger along the axial direction than laterally. The lateral direction generally has weaker flow and steeper slopes than the axial direction, therefore, slope movement becomes importance in material movement in the lateral direction, while, material movement along the axial direction can be well simulated by water flow alone.

3.3 Exploring Other Factors Regulating Transport from Shoal to Channel

The improvement of the DO simulation by slope movement in this paper is a preliminary study and further analysis is necessary. We can also explore other methods that may also promote drift of organic suspended solids to the channel, for example, adjustment of settling velocity.

A slower settling rate allows material to suspend at the water column for a longer time and have more horizontal movements before settling on the bed. If the dominant movement is toward the channel, then the channel may receive more volatile suspended solids under a slower settling than a faster settling rate. However, if the movement direction is dominantly unidirectional, the differences in material transport from the shoal to the channel between a slower settling and faster settling rate may be insignificant.

Ultimately a more sophisticated simulation of the resuspension of organic particulates from the bed and its subsequent transport is what's needed to more fully represent the movement of organic material from the shoals to the channel.

4 Conclusion

Besides movement due to hydrodynamic forces, in this simulation, particulate organic particles can settle down to the bed and move along the slope of the bed. This paper's approach in the simulation of organic particulate "focusing" to deep waters is important in the simulation of volatile suspended solid transport from shoal to channel, and for the simulation of anoxia and hypoxia in deep waters. The simulation of slope movement appears important in eutrophic simulations when the model does not adequately simulate the resuspension and transport of organic particles. Implementation of the slope movement for the Chesapeake Bay Estuarine Model significantly improves model simulations of summer anoxia.

References

1. Cerco, C.F., Cole, T.M.: Three-Dimensional Eutrophication Model of Chesapeake Bay. Technical Report EL-94-4, U.S. Army Engineer Waterways Experiment Station, Vicksbutg, MS, USA (1994)
2. D'Elia, C.F., Harding, L.W., Leffler, M., Mackiernan, G.B.: The role and control of nutrients in Chesapeake Bay. Water Sci. Tech. 26, 2635–2644 (1992)
3. Kemp, W.M., Boynton, W.R., Adolf, J.E., Boesch, D.F., Boicourt, W.C., Brush, G., Cornwell, J.C., Fisher, T.R., Gilbert, P.M., Hagy, J.D., Harding, L.W., Houde, E.D., Kimel, D.G., Miller, W.D., Newell, R.I.E., Roman, M.R., Smith, E.M., Stevenson, J.C.: Eutrophication of the Chesapeake Bay: Historic trends and ecological interactions. Marine Ecology Progress Series 303, 1–29 (2005)
4. Johnson, J.H., Kim, K.W., Heath, R.E., Hsieh, B.B., Butler, L.: Validation of a three-dimensional hydrohynamic model of Chesapeake Bay. J. Hydr. Engrg., ASCE 199(1), 2–20 (1993)
5. Postma, H.: Sediment transport and sedimentation in the estuarine environment. In: Lauff, G.H. (ed.) Estuaries, pp. 158–179. Am. Asso. Adv. Sci., Wash. (1967)

6. Cerco, C.F., Noel, M.: The 2002 Chesapeake Bay Eutrophication Model, US Army Corps of Engineers, prepared for USEPA Chesapeake Bay Program. EPA-903-R-04-004 (2004)
7. CBPO Modeling Subcommittee: Chesapeake Bay Modeling Subcommittee April meeting (2008), http://www.chesapeakebay.net/committee_msc_meetings.aspx
8. Johnson, J.H., Heath, R.E., Hsieh, B.B., Kim, K.W., Butler, L.: User's guide for a three-dimensional numerical hydrodynamic, salinity, and temperature model of Chesapeake Bay, U.S. Army Engineer, Baltimore, MD, USA (1991)
9. Donaldson, C.: Atmospheric turbulence and the dispersal of atmospheric pollutants. In: Haugen, D.A. (ed.) Workshop on Micrometeorology, pp. 313–390. American Meteorological Society, Boston (1973)
10. Sheng, Y.P.: A three-dimensional mathematical model of coastal, estuarine and lake currents using boundary fitted grid, Report No. 585, A.R.A.P. Group of Titan systems, New Jersey, Princeton, NJ (1986)
11. Harding, L.W., Mallonee, M.E., Perry, E.S.: Toward a predictive understanding of primary productivity in a temperate partially stratified estuary. Estuar. Coastal Shelf Sci. 55, 437–463 (2002)

Explicit Time Stepping Methods with High Stage Order and Monotonicity Properties

Emil Constantinescu[1] and Adrian Sandu[2]

[1] Mathematics and Computer Science Division,
Argonne National Laboratory, Argonne, IL 60439, USA
[2] Department of Computer Science, Virginia Tech, Blacksburg, VA 24061, USA

Abstract. This paper introduces a three and a four order explicit time stepping method. These methods have high stage order and favorable monotonicity properties. The proposed methods are based on multistage-multistep (MM) schemes that belong to the broader class of general linear methods, which are generalizations of both Runge-Kutta and linear multistep methods. Methods with high stage order alleviate the order reduction occurring in explicit multistage methods due to non-homogeneous boundary/source terms. Furthermore, the MM schemes presented in this paper can be expressed as convex combinations of Euler steps. Consequently, they have the same monotonicity properties as the forward Euler method. This property makes these schemes well suited for problems with discontinuous solutions.

1 Introduction

The numerical solution of time-dependent partial differential equations and nonlinear hyperbolic conservation laws are of great practical importance as they model diverse physical phenomena that appear in engineering, aeronautics, astrophysics, meteorology oceanography, environmental sciences, etc. Representative examples for nonlinear hyperbolic conservation laws include gas dynamics, shallow water flow, ground-water flow, non-Newtonian flows, traffic flows, advection and dispersion of contaminants, etc.

In the "method of lines" approach the temporal and spatial discretizations are independent. Traditionally Runge-Kutta (RK) and linear multistep methods (LM)s have been used for the integration of ODEs and semi-discrete time-dependent PDEs. General linear (GL) methods [1,2] represent a natural generalization of both Runge-Kutta (RK) and linear multistep (LM) methods. The methods investigated in this work are based on multistage-multistep (MM) schemes that belong to the broader class of GL methods. Multistep-multistage schemes are aimed at enhancing the stability and accuracy properties of the classical RK and LM methods. They use both internal stages like RK methods and information from previous solution steps like LM methods.

Explicit Runge-Kutta methods have stage order equal to one, and hence are subject to order reduction in the presence of non-homogeneous boundary and source terms [3,13,14]. The proposed high-stage order MM methods alleviate

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 293–301, 2009.

this problem without loosing their explicit character. Moreover, they can be expressed as convex combinations of Euler steps, and consequently, they have the same monotonicity properties that the spatial discretization method has with the forward Euler time stepping scheme, but with a different time step restriction [15,6]. This property makes the proposed MM schemes well suited for problems with discontinuous solutions (e.g., hyperbolic problems). Furthermore, the monotonicity properties can also guarantee the positivity of the solution.

In this study we investigate explicit time stepping methods of orders three and four based on MM schemes. The proposed methods have high stage order and favorable monotonicity properties. These features allow them to:

- avoid order reduction due to non-homogeneous boundary/source terms and
- prevent non-physical behavior with discontinuous solutions.

The proposed methods are aimed at modeling the transport components in atmospheric and oceanic simulations. The rest of this manuscript is organized as follows. In Sections 2 and 3 we present some background material on MM schemes and the monotonicity property considered in this study. The two proposed methods are presented in Sec. 4. Numerical experiments that illustrate the main features of the new methods are shown in Sec. 5. A short discussion concludes the paper.

2 Problem Formulation and Monotonicity Considerations

In this work we are concerned with the numerical solution of nonlinear time-dependent partial differential equations in the method of lines approach:

$$y'(t) = f(t, y(t))\,,\ t_0 < t < t_{\text{Final}}\,,\ y(t_0) = y_0\,, \tag{1}$$

where f represents the discretization of the spatial variables forming a semi-discrete equation, continuous in time. System (1) is nonauthonomous, however, for brevity we skip the time argument of f, unless noted otherwise.

We next introduce the concept of strong stability which defines the monotonicity properties that the proposed methods obey.

Definition 1 (Strong stability[12,6,15]). *A sequence* $\{y^{(n)}\}$ *is said to be strongly stable in a given semi-norm* $||\cdot||$ *if* $||y^{(n+1)}|| \leq ||y^{(n)}||$ *for all* $n \geq 0$.

Strong stability preserving (SSP) integrators are high order time stepping schemes that preserve the stability properties of the spatial discretization used with explicit Euler time stepping. Spurious oscillations can occur in a numerical solution that obeys the classical linear stability [6]. In PDEs with hyperbolic components an appropriate spatial discretization combined with an SSP time stepping method yields a numerical solution that does not exhibit nonlinear instabilities.

The favorable properties of SSP schemes derive from convexity arguments. In particular, if the forward Euler method is strongly stable for any time step smaller than Δt_{FE} (i.e., $||y + \Delta t f(y)|| \leq ||y||, \Delta t \leq \Delta t_{\text{FE}}$), then higher-order

methods can be constructed as convex combinations of forward Euler steps with various step sizes [15]. For example an explicit s-stage Runge-Kutta method can be represented in Euler steps:

$$y_{[n]} = y_{[n-1]}^{(s+1)}\,, \quad y_{[n-1]}^{(1)} = y_{[n-1]}\,, \tag{2a}$$

$$y_{[n-1]}^{(i)} = \sum_{j=1}^{i-1} \left[\alpha^{(i,j)} y_{[n-1]}^{(j)} + \beta^{(i,j)} \Delta t F_{[n-1]}^{(j)}\right] \;;\; i = 2,\, 3, \ldots,\, s, s+1\,. \tag{2b}$$

SSP methods preserve the strong stability of the forward Euler scheme for bounded time steps $\Delta t \le \mathcal{C} \cdot \Delta t_{\mathrm{FE}}$, where $\mathcal{C}$ is referred to as the CFL coefficient for the SSP property.

Theorem 1 (SSP for Runge-Kutta methods[6,15]). *If the forward Euler method is strongly stable under the CFL restriction $\Delta t \le \Delta t_{\mathrm{FE}}$, then the Runge-Kutta method (2) with $\beta^{(i,j)} \ge 0$ is SSP provided that $\Delta t \le \mathcal{C}\Delta t_{\mathrm{FE}}$, where*

$$\mathcal{C} = \mathrm{Min}\left\{\left(\alpha^{(\mathrm{i,j})}/\beta^{(\mathrm{i,j})}\right) :\; 1 \le \mathrm{i} \le \mathrm{s},\, 1 \le \mathrm{j} \le \mathrm{i}-1,\, \beta^{(\mathrm{i,j})} \ne 0\right\}.$$

In order to compare methods with different computational cost, a scaled or effective CFL coefficient, denoted by $\widehat{\mathcal{C}}$, is obtained by scaling the method's CFL with the number of right-hand-side evaluations.

3 Multistep Multistage Methods

We consider the following explicit k-step s-stage multistep-multistage method to compute the numerical solution of (1) with time step Δt. The solution at step n, $y_{[n]} \approx y(t_n) = y(n\Delta t)$ is given by

$$y_{[n]} = y_{[n-1]}^{(s+1)}\,, \quad y_{[n-1]}^{(1)} = y_{[n-1]}\,, \tag{3a}$$

$$y_{[n-1]}^{(i)} = \sum_{\ell=2}^{k}\sum_{j=1}^{s} \left[\alpha_{[n-\ell]}^{(i,j)} y_{[n-\ell]}^{(j)} + \beta_{[n-\ell]}^{(i,j)} \Delta t F_{[n-\ell]}^{(j)}\right] + \tag{3b}$$

$$+ \sum_{j=1}^{i-1} \left[\alpha_{[n-1]}^{(i,j)} y_{[n-1]}^{(j)} + \beta_{[n-1]}^{(i,j)} \Delta t F_{[n-1]}^{(j)}\right] \;;\; i = 2,\, 3, \ldots,\, s, s+1\,,$$

where $F_{[n-\ell]}^{(i)} = f\left(t_{[n-\ell]} + c_i \Delta t, y_{[n-\ell]}^{(i)}\right)$. We refer to $y_{[n-\ell]}^{(i)}$, $i = 1 \ldots s+1$, $\ell = 1 \ldots k$ as the stage i value at step $n-\ell$, and to $F_{[n-\ell]}^{(i)}$ as the corresponding stage derivative. The first sum in (3b) represents linear combinations of stage values and derivatives evaluated at previous steps, whereas the second sum describes the internal stages of the current step evaluation. Each stage value $y_{[n-\ell]}^{(i)}$ is an approximation to $y\left(t_{n-\ell} + c_i \Delta t\right)$. The *abscissa*, c, is determined from the consistency conditions.

The linear stability of method (3) is analyzed on a linear scalar test problem: $y'(t) = \lambda y(t)$, $\lambda \in \mathbb{C}$. By applying (3) to the test problem yields a solution of form $y^{n+1} = R(z)\, y^n$, where $z = \lambda \Delta t$ and $R(z)$ is referred to as the stability function of the method. Method (3) is linearly stable if $|R(z)| \leq 1$. The linear stability region is defined as the set $\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}$.

We give the following result without proof.

Theorem 2 (SSP for MM methods). *If the forward Euler method is strongly stable under the CFL restriction $\Delta t \leq \Delta t_{\rm FE}$, then method (3) with $\beta^{(i,j)}_{[n-\ell]} \geq 0$ is SSP provided that $\Delta t \leq \mathcal{C} \Delta t_{\rm FE}$, where*

$$\mathcal{C} = \text{Min}\left\{ \left(\alpha^{(\text{i,j})}_{[\text{n}-\ell]} / \beta^{(\text{i,j})}_{[\text{n}-\ell]} \right) : \, 1 \leq \text{i} \leq \text{s}, \, 1 \leq \text{j} \leq \text{i} - 1, \, 1 \leq \ell \leq \text{k}, \, \beta^{(\text{i,j})}_{[\text{n}-\ell]} \neq 0 \right\}.$$

By using consistency and convexity arguments the above theorem reduces to Theorem 1 and the proof is given in [6]. The importance of the SSP property is illustrated in Fig. 2.a where non-physical oscillations develop in the solution.

4 The Proposed Methods

In this section we present two new explicit multistep-multistage schemes that have stage order equal to three and are strong stability preserving.

4.1 Method MM p3 q3

Method MM p3 q3 (4) is an order three and stage order three ($p = 3$, $q = 3$) MM method with three stages and two steps ($s = 3$, $k = 2$). The CFL coefficient is $\mathcal{C}$=1.44 ($\widehat{\mathcal{C}}$=0.48).

$$\begin{array}{ll}
\alpha^{(2,1)}_{[n-1]} = 0.697169114587643 & \beta^{(2,1)}_{[n-1]} = 0.484471495618137 \\
\alpha^{(3,2)}_{[n-1]} = 0.76354468478889 & \beta^{(3,2)}_{[n-1]} = 0.530596705549337 \\
\alpha^{(4,3)}_{[n-1]} = 0.816170594740032 & \beta^{(4,3)}_{[n-1]} = 0.567167105426239 \\
\hline
\alpha^{(2,1)}_{[n-2]} = 0.302830885412357 & \beta^{(2,1)}_{[n-2]} = 0.109139040169882 \\
\alpha^{(3,1)}_{[n-2]} = 0.23645531521111 & \beta^{(3,1)}_{[n-2]} = 0.109233120743169 \\
\alpha^{(4,1)}_{[n-2]} = 0.183829405259968 & \beta^{(4,1)}_{[n-2]} = 0.106231031926622 \\
\hline
c = [0,\, 0.290779650375662,\, 0.625397767570505\,, 1]^T. &
\end{array} \tag{4}$$

4.2 Method MM p4 q3

Method MM p4 q3 (5) is an order four and stage order three ($p = 4$, $q = 3$) MM method with two stages and four steps ($s = 2$, $k = 4$). The CFL coefficient is $\mathcal{C}$=0.64 ($\widehat{\mathcal{C}}$=0.32).

$$
\begin{array}{l}
\alpha^{(2,1)}_{[n-1]} = 0.641788036235959\ \beta^{(2,1)}_{[n-1]} = 1. \\
\alpha^{(3,2)}_{[n-1]} = 0.530533524263627\ \beta^{(3,2)}_{[n-1]} = 0.826649133840462 \\
\hline
\alpha^{(3,1)}_{[n-2]} = 0.278475821635639\ \beta^{(3,1)}_{[n-2]} = 0.433906221232917 \\
\hline
\alpha^{(2,1)}_{[n-3]} = 0.295361832953222\ \beta^{(2,1)}_{[n-3]} = 0.354153138170544 \\
\alpha^{(3,1)}_{[n-3]} = 0.111760513607703\ \beta^{(3,1)}_{[n-3]} = 0.174139291008244 \\
\hline
\alpha^{(2,1)}_{[n-4]} = 0.062850130810818 \\
\alpha^{(3,1)}_{[n-4]} = 0.07923014049303 \\
\hline
c = [0,\ 0.574879079831644\,,1]^T.
\end{array}
\tag{5}
$$

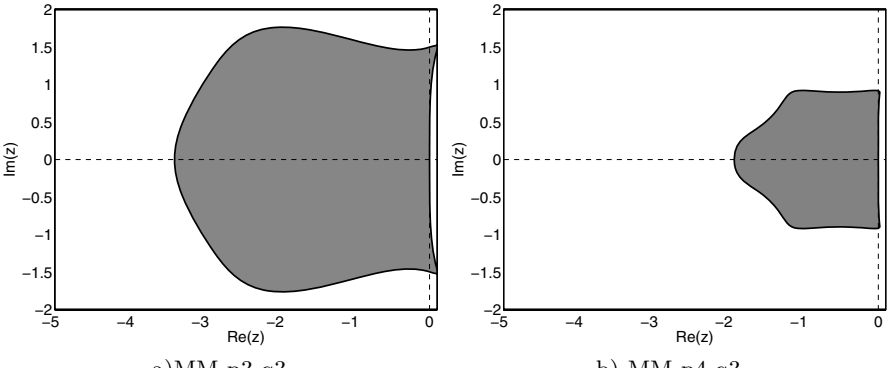
4.3 Linear Stability

The linear stability region for MM p3 q3 (4) is shown in Figure 1.a. We remark that the stability region contains a segment of the imaginary axis, which is a desirable property when solving PDEs via the method of lines with certain spatial discretizations [9].

In Figure 1.b we show the stability region of MM p4 q3 (5), and here we note again that the stability region contains a segment of the imaginary axis. The region is smaller than in the case of (4); however, the fourth order method requires only two function evaluation. It follows that MM p4 q3 has two thirds of the cost of MM p3 q3.

4.4 Starting Procedures

Each step of the MM method (3) requires past precomputed information, specifically, $y^{(j)}_{[n-\ell]}$ and $F^{(j)}_{[n-\ell]}$, $2 \le \ell \le k$. In this study the initial step is considered to provide an approximation to the exact solution and its derivative at the corresponding time within order p, the order of the MM method under consideration.



a)MM p3 q3 b) MM p4 q3

Fig. 1. Linear stability regions (shaded) for MM p3 q3 and MM p4 q3

In practice, for the proposed methods, one can easily compute the initial solution and its derivative components with the classical SSP RK schemes [6,8] of corresponding orders and at the respective times as given by the method abscissa.

5 Numerical Results

In this section we present two numerical experiments that illustrate the properties of the two proposed MM methods. In the first experiment we investigate the SSP (monotonicity) properties. In the second numerical experiment we present the order reduction phenomenon, and show how it can degrade the accuracy of high order (p) low stage order ($q = 1$) multistage methods. We further show that the proposed methods maintain their corresponding orders of consistency (p).

5.1 Monotonicity

Methods with SSP properties are needed to evolve in time solutions that may develop discontinuities of hyperbolic PDEs. The SSP conditions impose a very strict restriction on the time steps, and hence the time stepping scheme efficiency is very important.

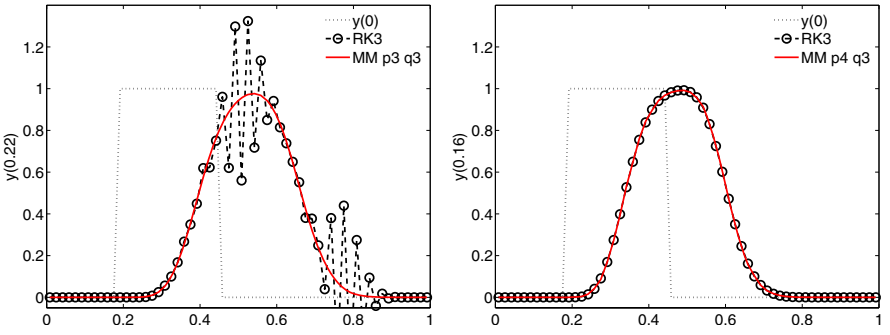
Figure 2 shows the solutions of the advection equation obtained with the proposed methods MM p3 q3 (4) and MM p4 q3 (5) and the optimal third order RK scheme, RK3, with three stages, $\mathcal{C} = 1$ ($\widehat{\mathcal{C}} = 0.33$) [6,8]. The space discretization is first order upwind, chosen for its well understood behavior. The time step for MM p3 q3 is such that the CFL coefficient is 1.3. At $t = 0.22$ (Figure 2.a) the MM method solution remains oscillation free, while the RK3 solution shows the effects of linear instability. The solution obtained by using MM p4 q3 (Figure 2.b) is also stable, but at a lower CFL coefficient comparable, however, with the one used for the MM p3 q3 case; i.e., MM p3 q3 and RK3 require three function evaluations per step, and hence, a CFL of two thirds is needed for a fair comparison.

We next explore the monotonicity properties of the SSP MM methods on a nonlinear hyperbolic equation. The inviscid Burgers' equation is

$$\frac{\partial y(t,x)}{\partial t} + \frac{\partial}{\partial x}\left(\frac{1}{2}y(t,x)^2\right) = 0\,, \quad 0 \le x \le 1\,,\ 0 \le t \le t_{\text{Final}}\,. \tag{6}$$

The spatial discretization uses the third-order upwind-biased flux limited scheme based [4,10,11]. This spatial discretization is SSP with forward Euler steps and hence, with the proposed MM methods described in this work. The SSP condition is satisfied if the CFL coefficient of the method $\mathcal{C}$ is smaller than the CFL number of the problem: $\mathcal{C} \le$ problem CFL number $= \max(y)\Delta t/\Delta x$.

In Figure 3 we show the solution of the Burgers' equation integrated with RK3 ($s = 3, \mathcal{C} = 1$) and MM p3 q3 (4) ($\mathcal{C}$=1.44) at time 0.25 with a method CFL of 1.5.



a) MM p3 q3, $\mathcal{C} = 1.3$ b) MM p4 q3, $\mathcal{C} = 0.86$

Fig. 2. Solution of the advection equation integrated in time with RK3 and the multistep-multistage schemes MM p3 q3 and MM p4 q3

The solution given by the MM scheme remains oscillation free, whereas the classical RK method becomes unstable.

5.2 Avoiding Order Reduction

Order reduction describes the behavior where the effective order of a numerical method on a given problem is smaller than its theoretical order as given by the classical theory. Order reduction can considerably degrade the efficiency of the numerical integration. Moreover, order reduction is difficult to detect in practical computations because embedded methods used for error estimation are also affected by it.

Explicit RK methods have the stage order equal to one, which makes them susceptible of order reduction for problems with non-homogeneous boundary conditions and/or nonzero source terms. In order to illustrate this, we consider the test problem from [14] (advection with a nonlinear source term):

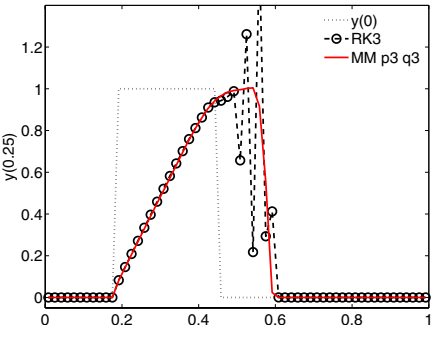
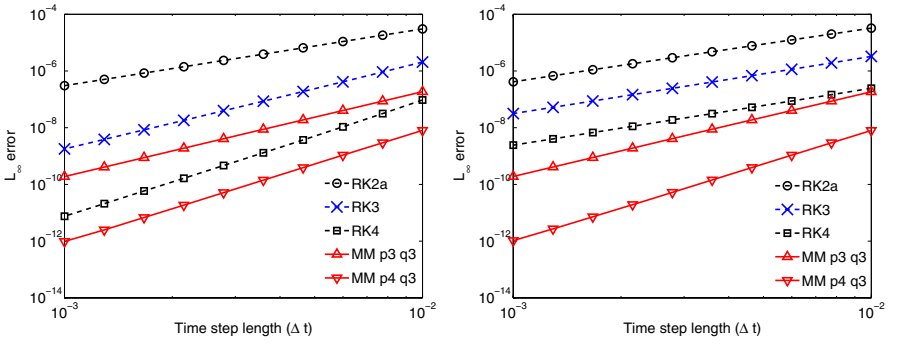


Fig. 3. Solution of Burgers' equation integrated in time with RK3 ($\mathcal{C} = 1.00$) and the multistep-multistage schemes MM p3 q3 ($\mathcal{C} = 1.44$). The CFL of the problem is 1.5.

$$\frac{\partial y(t,x)}{\partial t} = -\frac{\partial y(t,x)}{\partial x} + b(t,x)\,, \quad \begin{matrix} 0 \le x \le 1 \\ 0 \le t \le 1 \end{matrix}\,, \quad \begin{matrix} y(t,0) = b(t,0) \\ y(0,x) = y_0(x) \end{matrix}\,.$$

The initial condition is $y_0(x) = 1 + x$ and the (left) boundary and source term is $b(t,x) = (t-x)/(1+t)^2$. The exact solution given by $y(t,x) = (1+x)/(1+t)$ is



a) Error vs. time step - no forcing b) Error vs. time step - with forcing

Fig. 4. Numerical illustration of the order reduction phenomenon. The L_∞ norm of error is shown versus the time step length for RK methods of orders 2-4 and MM p3 q3 and MM p4 q3. (a) Results with no forcing show that the effective order of each method equals its theoretical order. (b) When forcing is present the effective order of RK methods is two (order reduction). The high stage-order MM methods maintain their theoretical orders of accuracy.

linear in space, allowing us to use first order upwind space discretization without introducing discretization errors. For the time integration we employ the typical RK methods of orders 2, 3, and 4. Sanz-Serna et al. [14] show that RK methods with $p \geq 3$ suffer from order reduction. This theoretical result is verified in our numerical experiment.

Figure 4.a shows the discretization error versus the time step with the forcing terms switched off [14]. In this case all methods retain their expected order, verifying the classical theory. In Figure 4.b we show the results with stiff boundary and source terms. In this case both the third order RK3a method [5] and the fourth order "classical" RK4 method [7] display second order behavior. In these situations a second order method can be more efficient than higher order methods. The high stage order proposed methods MM p3 q3 (4) and MM p4 q3 (5) retain their corresponding orders of consistency.

6 Discussion

In this paper we introduce two new explicit multistage-multistep methods with high stage orders for solving ordinary differential equations and PDEs via the method of lines. The MM methods are SSP – they have the monotonicity properties of forward Euler scheme, but under a different time step restriction.

To our knowledge the proposed methods are the first explicit high-stage order SSP methods. The two numerical experiments presented in this paper motivate both properties – SSP and high stage order.

An error control mechanism can be considered by using a lower order embedded method; however, changing the time step requires restarting the method.

Acknowledgement

This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357.

References

1. Burrage, K., Butcher, J.C.: Non-linear stability of a general class of differential equation methods. BIT 20(2), 185–203 (1980)
2. Butcher, J.C.: General linear methods for ordinary differential equations. Mathematics and Computers in Simulation (2007) (in press)
3. Carpenter, M.H., Gottlieb, D., Abarbanel, S., Don, W.-S.: The theoretical accuracy of Runge–Kutta time discretizations for the initial boundary value problem: A study of the boundary error. SIAM Journal on Scientific Computing 16(6), 1241–1252 (1995)
4. Chakravarthy, S., Osher, S.: Numerical experiments with the Osher upwind scheme for the Euler equations. AIAA Journal 21, 1241–1248 (1983)
5. Gottlieb, S.: On high order strong stability preserving Runge–Kutta and multi step time discretizations. Journal of Scientific Computing 25(1), 105–128 (2005)
6. Gottlieb, S., Shu, C.-W., Tadmor, E.: Strong stability-preserving high-order time discretization methods. SIAM Rev. 43(1), 89–112 (2001)
7. Hairer, E., Norsett, S.P., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Springer, Heidelberg (1993)
8. Higueras, I.: On strong stability preserving time discretization methods. Journal of Scientific Computing 21(2), 193–223 (2004)
9. Hundsdorfer, W., Verwer, J.G.: Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations, vol. 33. Springer, Heidelberg (2003)
10. Osher, S., Chakravarthy, S.: High resolution schemes and the entropy condition. SIAM Journal on Numerical Analysis 21(5), 955–984 (1984)
11. Osher, S., Chakravarthy, S.: Very high order accurate TVD schemes. Oscillation Theory, Computation, and Methods of Compensated Compactness, IMA Vol. Math. Appl. 2, 229–274 (1986)
12. Ruuth, S.J., Hundsdorfer, W.: High-order linear multistep methods with general monotonicity and boundedness properties. Journal of Computational Physics 209(1), 226–248 (2005)
13. Sanz-Serna, J.M., Verwer, J.G.: Stability and convergence at the PDE/stiff ODE interface. Appl. Numer. Math. 5(1-2), 117–132 (1989)
14. Sanz-Serna, J.M., Verwer, J.G., Hundsdorfer, W.: Convergence and order reduction of Runge-Kutta schemes applied to evolutionary problems in partial differential equations. Numer. Math. 50(4), 405–418 (1987)
15. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. Journal of Computational Physics 77(2), 439–471 (1988)

Improving GEOS-Chem Model Tropospheric Ozone through Assimilation of Pseudo Tropospheric Emission Spectrometer Profile Retrievals

Kumaresh Singh[1], Paul Eller[1], Adrian Sandu[1], Kevin Bowman[3], Dylan Jones[2], and Meemong Lee[3]

[1] Department of Computer Science, Virginia Tech, Blacksburg, VA 24060
{kumaresh,peller,sandu}@cs.vt.edu
[2] Department of Physics, University of Toronto, Toronto, ON, Canada
dbj@atmosp.physics.utoronto.ca
[3] NASA Jet Propulsion Laboratory, Padadena, CA 91109, USA
{kevin.bowman,Meemong.Lee}@jpl.nasa.gov

Abstract. 4D-variational or adjoint-based data assimilation provides a powerful means for integrating observations with models to estimate an optimal atmospheric state and to characterize the sensitivity of that state to the processes controlling it. In this paper we present the improvement of 2006 summer time distribution of global tropospheric ozone through assimilation of pseudo profile retrievals from the Tropospheric Emission Spectrometer (TES) into the GEOS-Chem global chemical transport model based on a recently-developed adjoint model of GEOS-Chem v7. We are the first to construct an adjoint of the linearized ozone parameterization (linoz) scheme that can be of very high importance in quantifying the amount of tropospheric ozone due to upper boundary exchanges. Tests conducted at various geographical levels show that the mismatch between adjoint values and their finite difference approximations could be up to 87% if linoz module adjoint is not used, leading to a divergence in the quasi-Newton approximation algorithm (L-BFGS) during data assimilation. We also present performance improvements in this adjoint model in terms of memory usage and speed. With the parallelization of each science process adjoint subroutine and sub-optimal combination of checkpoints and recalculations, the improved adjoint model is as efficient as the forward GEOS-Chem model.

Keywords: Global chemistry and transport, Adjoints, Inverse modeling, Data assimilation, Linoz scheme.

1 Introduction

GEOS–Chem is a global 3-D model of atmospheric composition driven by assimilated meteorological observations from the Goddard Earth Observing System (GEOS) of the NASA Global Modeling and Assimilation Office. It is applied by research groups around the world to study a wide range of atmospheric composition problems such as assessing intercontinental transport of pollution, evaluating consequences of regulations and climate change on air quality, comparison of model estimates to

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 302–311, 2009.

satellite observations and field measurements, and fundamental investigations of tropospheric chemistry.

Adjoint models are powerful tools widely used in meteorology and oceanography for applications such as data assimilation, model tuning, sensitivity analysis, and the determination of singular vectors. The adjoint model computes the gradient of a response function with respect to control variables. Generation of adjoint code may be seen as the special case of differentiation of algorithms in reverse mode, where the dependent function is a scalar. Developing a complete adjoint of global atmospheric models involves rigorous work of constructing and testing adjoints of each of the complex science processes individually, and integrating those into a consistent adjoint model. It is well accepted that adjoint construction is an extremely challenging task.

Original work on the adjoint of GEOS-Chem began in 2003, focusing on the adjoint of the offline aerosol simulation. By 2005, the adjoint was expanded to include a tagged CO simulation and a full chemistry simulation as well as observational operators for MOPITT (CO) and IMPROVE network (aerosols). Henze et. al(2007) [1] discusses the construction of adjoint for GEOS-3 v6 of GEOS-Chem. We implemented a standard adjoint model for GEOS-4 v7 of GEOS-Chem and made it more user friendly, with the wider goal of making this adjoint publicly available as part of the standard GEOS-Chem code. This will provide the community of GEOS-Chem users with the tools needed for performing sensitivity analysis and data assimilation.

KPP chemistry was first interfaced with GEOS-Chem and its adjoint in Henze et al. (2007), see Appendices therein. We improved upon this implementation in terms of automation, performance, benchmarking, and documentation. GEOS-4 convection and advection processes are based on entirely different algorithms as compared to GEOS-3. Convection and wet-deposition adjoints are discrete adjoints and were constructed in a hybrid fashion using the automatic differentiation software TAMC [2] together with manual coding. Advection adjoint on the other hand is continuous and was obtained by calling the forward subroutine with reverse wind fields. We also provided a way of calculating the scaled emission and dry-deposition adjoints by modifying the KPP chemistry integrator. All these pieces are tested extensively and integrated together to build the full adjoint GEOS-Chem model.

In this paper we present 4-D variational data assimilation [3] results of forecast improvements through pseudo profile retrievals from Tropospheric Emission Spectrometer (TES, JPL NASA) [4]. The designed interface to run 4-D Var data assimilation using GEOS-Chem allows users to tweak all the limited memory BFGS (L-BFGS) method [5] and GEOS-Chem related parameters through a single driver file in addition a list of plug-n-play response function calculation subroutines including TES observation operator based forcing. We also present the first validation results of the adjoint of linearized ozone chemistry (linoz) scheme [6]. The construction of linoz scheme adjoint is a major step towards computing the sensitivity of tropospheric ozone with respect to upper boundary layer exchanges. The standard version of GEOS-Chem adjoint is completely parallel. The continuous advection adjoint inherits the parallelism of the forward subroutine. For rest of the science process adjoints, we have implemented shared memory system based parallel version. Some discrete adjoint calculations require intermediate variable values from forward calculations. In such cases, either these variables are recomputed in the adjoint mode or are written to checkpoint files [7] during the forward calculation and read in the adjoint mode.

We recalculated variable values in the adjoint mode wherever this had a minimal performance penalty.

This paper is organized as follows. Section 2 discusses the construction of adjoints for various science processes. It also summarizes software engineering aspects of the standard GEOS-Chem adjoint model and provides speedup results from shared memory based parallelization. Sections 3 exhibits validation results of the linoz scheme adjoint (verification of accuracy against finite difference approximations). Section 4 describes the framework for using the adjoint model in scientific applications such as 4-D variational data assimilation and displays improvement results from TES profile retrievals. Section 5 draws conclusions and points to future work.

2 Standard Adjoint Model of GEOS-Chem

The mathematical formulation for calculating gradients of a model output using the adjoint method can be derived from the equations governing the forward model or from the forward model code. The former approach leads to the continuous adjoint, while the latter leads to the discrete adjoint [8]. Continuous adjoint gradients may differ from the actual numerical gradients of cost function *J*, and continuous adjoint equations (and requisite boundary/initial conditions) for some systems are not always readily derivable; however, solutions to continuous adjoint equations can be more useful for interpreting the significance of the adjoint values. Many previous studies have also described the derivation of discrete adjoints of such systems [9][10]. An advantage of the discrete adjoint model is that the resulting gradients of the numerical cost function are exact, even for nonlinear or iterative algorithms, making them easier to validate. Furthermore, portions of the discrete adjoint code can often be generated directly from the forward code with the aid of automatic differentiation tools.

We have implemented the chemistry simulations in GEOS-Chem using the Kinetic Pre-Processor (KPP) [11]. KPP provides a library of several chemical solvers together with their tangent linear and adjoint integrators in addition to the native SMVGEARII solver [12][13]. It generates chemistry adjoint files in a similar fashion as it generates the forward chemistry. The generated files are interfaced with the GEOS-Chem adjoint code, updating the KPP global variables, parameters and initialization files. The adjoint of GEOS-4 advection subroutine is continuous and is derived by calling the forward subroutine with reversed wind fields. The GEOS-4 convection, planetary boundary mixing (pbl_mix) and wet deposition adjoints are discrete and have been constructed using the tangent linear and adjoint model complier (TAMC). Intermediate parameters in the forward run are checkpointed every dynamic time step and are read in during the adjoint run. The linoz scheme is linear and is self adjoint. In GEOS-Chem emission and dry deposition are handled through chemistry via fake equations. The rates for these processes are calculated separately and then attached to the chemistry reaction rates. The adjoints of these subroutines are scaled and are calculated using the adjoint integrator. The adjoint integrator provides adjoints with respect to the rates which are then multiplied with the individual rates and accumulated over time.

A detailed GEOS-Chem adjoint function call flow is presented in Figure 1. It provides a visualization of the order in which individual science processes are called in the forward and adjoint mode, and the way checkpoint files are written and read.

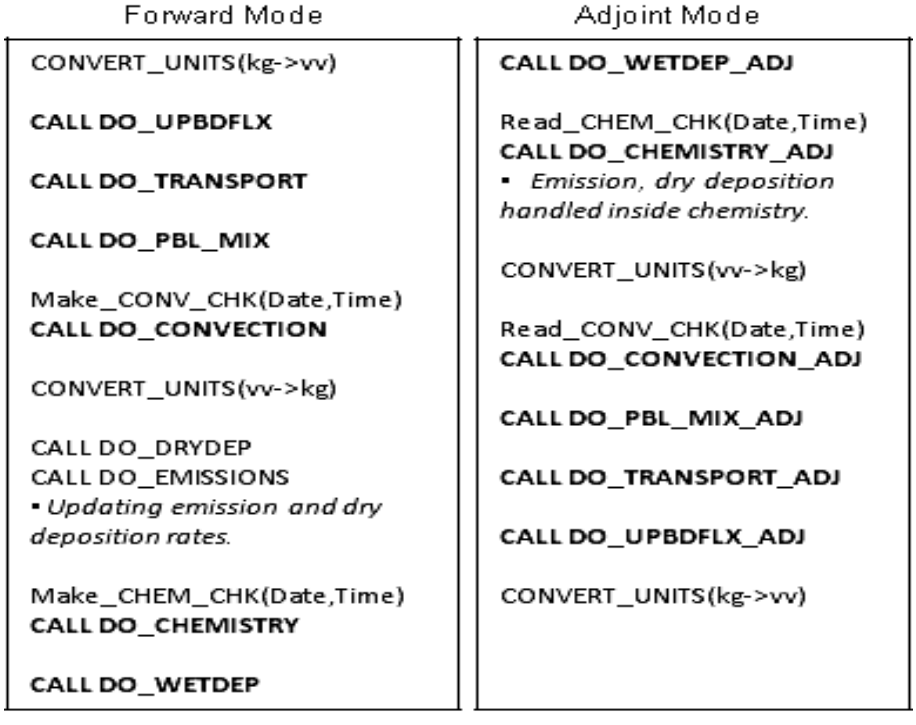


Fig. 1. GEOS-Chem forward and adjoint function call flows. The adjoint of science processes are called in reverse order in the adjoint mode. CONVERT_UNITS() converts unit for tracer variable in the forward mode, and adjoint variable in the adjoint mode. Make_*_CHK() are subroutines to make checkpoint files and Read_*_CHK() to read the checkpoint files as per input date and time.

2.1 Code Structure Overview

The newly developed GEOS-Chem adjoint model is well structured and follows the coding style provided in the GEOS-Chem users' manual [14]. For one science process, all the forward and related adjoint subroutines are kept in the same module file for the ease of users to look into only one file. To handle checkpointing, the additional module file CHECKPOINT_MOD.F has been provided. In addition, various subroutines to perform observation and background cost function calculations, define adjoint variables, include satellite observations are provided through separate files.

The standard GEOS-Chem adjoint package (GCv7_ADJ) is available for download from our project website [15]. Users have been provided with five modes of application built on top of the full adjoint model. The source files for each of these modes are in separate directories inside the main code directory. FWD_SMV mode is the forward

GEOS-Chem code that is available from the Harvard's website. This mode uses the SMVGEAR integrator for chemistry calculations. FWD_KPP is equivalent to FWD_SMV except it uses KPP for chemistry, providing users a suite of fast and highly accurate integrators to choose from. ADJ_FD is the finite difference testing module which users can choose to validate newly built adjoint subroutines. ADJ_SENST and ADJ_4DVAR are selected to perform sensitivity analysis and 4-D variational data assimilation respectively. Users can choose one of these options by simply (un)commenting the mode option in the compilation script (v7-04-10.cmp).

4-D variational data assimilation requires multiple iterations of forward and backward model runs before it converges to a suitable initial condition. Users have been provided the option to stop at any iteration and restart from the same point at a later time. The number of iterations to run is handled via the same run script (v7-04-10.run) which is used to initiate the geos executable.

2.2 Speedup Results: Shared Memory System Based Parallelization

Harvard's GEOS-Chem code is programmed parallel for shared memory systems. One of the challenges in developing the adjoint model was to parallelize this model completely. For chemistry adjoint we used THREADPRIVATE variables to allow multiple threads to execute the KPP chemistry routines for different grid cells in parallel. Emission and dry deposition adjoints are handled through chemistry. Advection adjoint being continuous leverages forward parallelization. For convection, planetary boundary mixing and wet deposition adjoints, we created OpenMP parallel versions of the subroutines taking care of the thread shared and private variables. The new GEOS-Chem adjoint code is completely parallel and has been tested for consistency against the serial version.

Discrete adjoints of non-linear (in terms of dependent input variable) subroutines require intermediate variable values from forward calculations. In such cases, either these variables are recomputed in the adjoint mode or are written to checkpoint files during the forward calculation and read in the adjoint mode. Both these options add performance penalty to the adjoint calculations. Checkpointing was used whenever recalculation involved lots of parameters and the required intermediate variables were few. The benefit of recalculations however is that these code portions could be parallelized in a similar way as the forward mode.

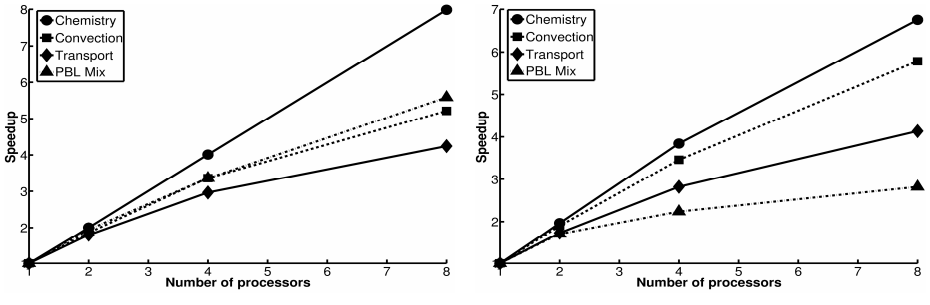


Fig. 2. Speedup graphs for chemistry, convection, advection and planetary boundary mixing subroutines in forward(*left*) and adjoint(*right*) mode on 1, 2, 4 and 8 processors. The simulation window for this analysis was 24 hours performed on July 2001 GEOS-Chem data.

Presented in Figure 2 are the speedup graphs for all the science process subroutines in forward and adjoint mode that are most time consuming. These graphs show that the adjoint mode scales well considering the performance penalties due to checkpointing and recalculations.

3 Linearized Ozone Chemistry Adjoint

The linearized ozone (linoz) scheme [6] is a stratospheric ozone chemistry mechanism for atmospheric models that focus on the troposphere and was developed with primary goals of (1) accurate calculation of the cross tropopause flux, and (2) reasonable representation of the ozone gradients near the tropopause. Earlier, in tropospheric CTMs, stratospheric ozone were handled by specifying climatology-based mixing ratios for ozone in the lower stratosphere and then allowing the transport to determine the cross-tropopause flux. The stratospheric level at which the ozone mixing ratios were specified, as well as other factors affecting the flux, varied from model to model. The fluxes produced by models using variants of this technique ranged from over 1400 Teragram(Tg)/yr, to less than 400 Tg/yr. Such discrepancies were important to resolve since they gave almost opposite interpretations of the role of tropospheric photochemistry: the large fluxes resulted in net photochemical loss of ozone throughout most of the troposphere, whereas the much smaller fluxes required a net production in order to balance with near-surface losses.

The developed linear model for ozone (Linoz) chemistry was simple and computationally efficient. In this method the ozone chemical tendency is expressed as a linear function of ozone, temperature, and the overhead ozone column. The linearizations are performed about an observed climatological state for a standard set of latitudes, months, and altitudes. The 24-hour, zonal-mean ozone photochemical tendency is calculated at each time step for each stratospheric grid box from these monthly varying coefficients. The change in ozone with time due to local chemistry is given by,

$$df/dt = (P\text{-}L)[\, f, T, Co_3] \,. \tag{1}$$

where (P - L) represents the ozone tendency (in units of ppmv/s), the square brackets denote a functional dependence, f is the ozone mixing ratio, T is temperature, and Co_3 is the column ozone above the point under consideration.

3.1 Linoz Adjoint Validation Results

The linoz scheme is linear and it is self adjoint, i.e. the adjoint variable follows the same dynamics as the forward tracer concentrations. Presented below in Figure 3 are the plots of mismatch between the adjoint values and their finite difference approximations with and without using linoz adjoint subroutine. These are generated using July 2006 summertime GEOS-Chem data run over a 1 day period at various geographical levels. The results show clearly that the consistency could be compromised by as large as 87% if the linoz adjoint subroutine is not used.

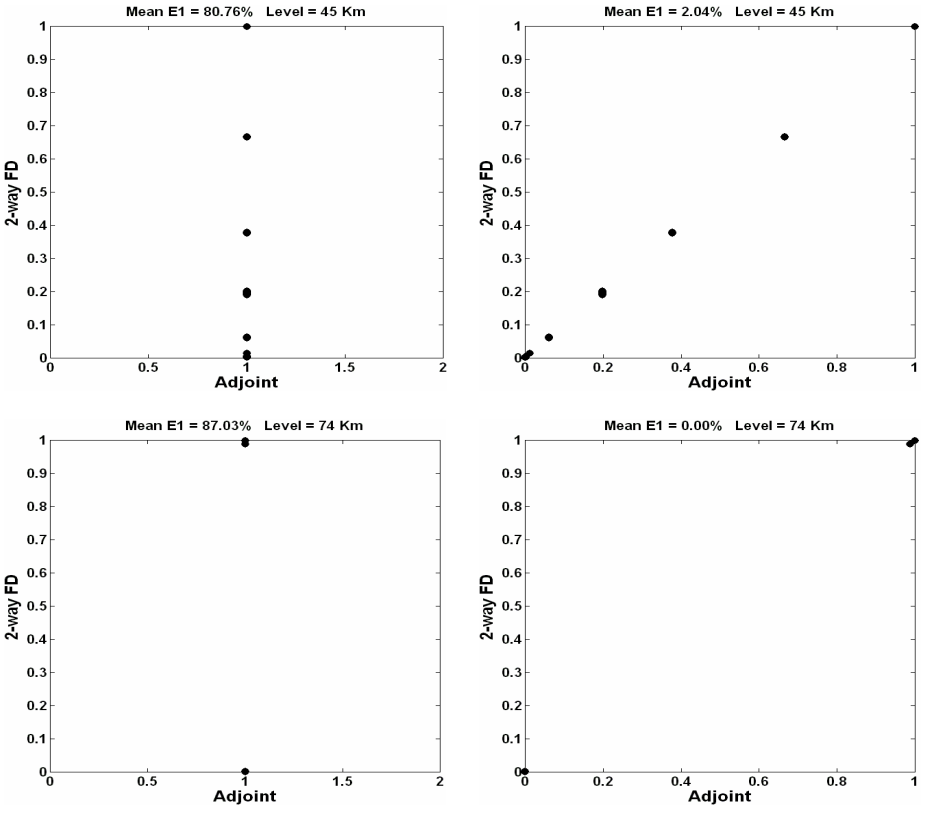


Fig. 3. Validation results of adjoint values with respect to their finite-difference approximations with (*right*) and without (*left*) using linoz adjoint subroutine for levels 45 Km (*top*) and 74 Km (*bottom*) for 2006 summertime GEOS-Chem data, 1 day simulation

4 Data Assimilation through TES Profile Retrievals

4D-Var data assimilation allows the optimal combination of three sources of information: an a priori (background) estimate of the state of the atmosphere, knowledge about the physical and chemical processes that govern the evolution of pollutant fields as captured in the chemistry transport model GEOS-Chem, and observations of some of the state variables.

The adjoint method has been used widely for variational data assimilations and inverse modeling. In these applications, a cost function is defined as

$$J = J_{Observations} + J_{Background}$$
$$= (\, y - F(x)\,)^T S_e^{-1} (\, y - F(x)\,) + (\, E - E^b\,)^T B^{-1} (\, E - E^b\,) . \quad (2)$$

where E is the state vector at the initial time, E^b is an *a priori* (background or initial guess) estimate of inputs, and B is background error covariance matrix, $y \in R^m$ is the observation vector whose vectors are retrievals of Ozone, m is the total number of

observations used in the inversion analysis, $S_e \in R^{mxm}$ is the block diagonal error covariance matrix of the observation vector, $x \in R^n$ is the state vector whose elements are the strengths of O_3 and $F(x) \in R^m$ is the TES observation operator applied to the state vector according the following equation,

$$F(x) = y_a + A_{yy} (\ln[H(x)] - y_a) . \quad (3)$$

where $H : R^n \rightarrow R^m$ represents the interpolation operator to convert GEOS-Chem vector into TES observation vector, A_{yy} is the averaging kernel and y_a is the *a priori* TES observation vector.

The cost function consists of two parts: the first part is a measure of model prediction errors, and the second part is a penalty for deviation from a priori estimates of model inputs. In typical applications the adjoint method is used to calculate the gradient of the cost function with respect to initial concentrations (for data assimilation applications) or model parameters such as emissions (for inverse modeling applications). This gradient is then used in an iterative optimization algorithm in order to minimize the cost function, reducing the mismatch of model predictions and observations by adjusting the inputs (e.g. emissions) within a reasonable range. Variational methods provide an important approach for constraining emissions of various species on a spatially resolved basis.

In order to carry out data assimilation using GEOS-Chem, an interface has been developed to call the optimization routine L-BFGS [16] through a driver file named 4dvar_driver.f. The framework provides users the ability to perform 4-D variational data assimilation with respect to both tracer and emission species. Below we present details of an experiment based on synthetic observations (*twin experiment framework*).

Framework Details. An observation grid is defined over the computational domain and synthetic observations are produced by performing a forward GEOS-Chem run on the initial reference concentration field c_0^0. A perturbation is then added to c_0^0 to get a perturbed initial condition c_p^0 (which is considered the background state)

$$c_p^0 = c_0^0 + \text{perturbation}$$

This perturbed concentration is then transferred to the optimization subroutine in order to obtain the best estimate c_{op}^0 of the original concentration c_0^0 after several iterations.

At iteration 0, $x_0 = c_p^0$

At each subsequent iteration k (k≥1),

$$x_{k+1} \leftarrow \text{L-BFGS} (x_k, f, g)$$
$$c_{op}^0 \leftarrow x_{k+1}$$
$$(f, g) \leftarrow \text{reverse\_mode} (c_{op}^0, \text{Observation\_Chk})$$

where f is the cost function and g is the gradient of the cost function.

4.1 Results

To validate the developed data assimilation test-bed, we conducted a twin experiment using ozone concentrations at 00:00 hrs on July 2006 as the initial condition with TES profiles generated from observations as twice the initial condition. Figure 4 represents

the correction in the initial condition after 17 iterations of the optimization routine, verified by the 32% decrease in the cost function within 9 model runs. (Intermediate iterations are of negligible cost as compared to the model runs that heavily dominate the cost of running l-BFGS optimization routine). The plots validate that the designed data assimilation framework works nicely.

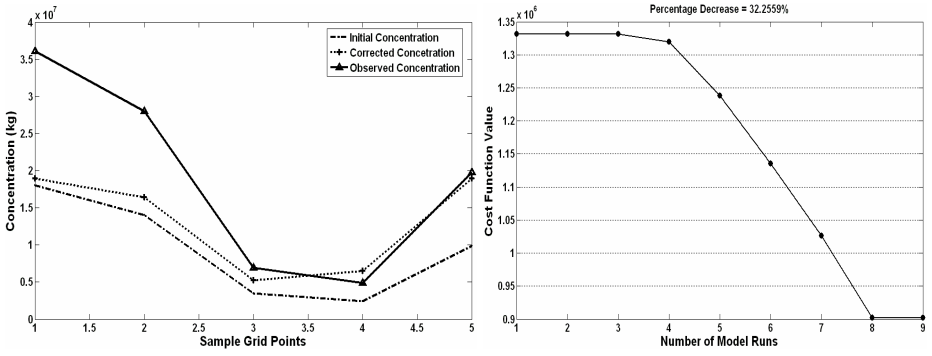


Fig. 4. 4-D variational data assimilation results for twin experiment run over 2006 summertime GEOS-Chem data for 3 days with TES profile retrievals generated synthetically. Plots include correction in the initial concentration (*left*) and decrease in the cost function with respect to model runs (*right*).

5 Conclusion

We have improved upon the standard adjoint model of GEOS-Chem for faster, more accurate and ready for real data experiments. The adjoint code has been fully parallelized and is as efficient as the forward mode considering the performance penalties added due to checkpointing and intermediate variable recalculations in the adjoint mode. It will allow world-wide community of GEOS-Chem users to perform powerful analytic and scientific applications such as sensitivity analysis and 4-D variational data assimilation in real time.

An adjoint of linoz chemistry will help scientists worldwide to better quantify the tropospheric ozone coming from upper boundary layer exchanges. We have also successfully implemented the TES observation operator based cost function calculations, ready to perform 4-D variational data assimilation and sensitivity calculations with real data. These improvements could add ample scientific value to the way we analyze physical and chemical dynamics of the earth's atmosphere.

References

1. Henze, D.K., Hakami, A., Seinfeld, J.H.: Development of the Adjoint of GEOS Chem. Atmos. Chem. Phys. 7, 2413–2433 (2007)
2. Giering, R., Kaminski, T.: Recipes for Adjoint code Construction. ACM Trans. Math. Softw. 24, 437–474 (1998)

3. Elbern, H., Schmidt, H.: A four-dimensional variational chemistry data assimilations scheme for Eulerian chemistry transport modeling. J. Geophys. Res. 104, 18583–18598 (1999)
4. Tropospheric Emission Spectrometer (TES), JPL NASA, http://tes.jpl.nasa.gov/
5. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. Scientific Computing 16, 1190–1208 (1995)
6. McLinden, C.A., Olsen, S.C., Hannegan, B., Wild, O., Prather, M.J., Sundet, J.: Stratospheric ozone in 3-D models: A simple chemistry and the cross-tropopause flux. J. Geophys. Res. 105(D11), 14653–14666 (2000)
7. Griewank, A., Walther, A.: Algorithm 799: Revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation. ACM Trans. Math. Softw. 26, 19–45 (2000)
8. Giles, M., Pierce, N.: An introduction to the adjoint approach to design. Flow, Turbulence and Control 65, 393–415 (2000)
9. Sandu, A., Daescu, D., Carmichael, G.R., Chai, T.: Adjoint sensitivity analysis of regional air quality models. J. Comput. Phys. 204, 222–252 (2005a)
10. Muller, J.F., Stavrakou, T.: Inversion of CO and NOx emissions using the adjoint of the IMAGES model. Atmos. Chem. Phys. 5, 1157–1186 (2005)
11. Sandu, A., Daescu, D.N., Carmichael, G.R.: Direct and adjoint sensitivity analysis of chemical kinetic systems with KPP: Part I – theory and software tools. Atmos. Environ. 37, 5083–5096 (2003)
12. Jacobson, M.Z., Turco, R.: SMVGEAR: A Sparse-Matrix, Vectorized Gear Code For Atmospheric Models. Atmos. Environ. 28, 273–284 (1994)
13. Jacobson, M.Z.: Technical Note: Improvement of SMVGEAR II on Vector and Scalar Machines through Absolute Error Tolerance Control. Atmos. Environ. 32, 791–796 (1998)
14. GEOS-Chem users manual, http://www-as.harvard.edu/chemistry/trop/geos/doc/man
15. GEOS-Chem adjoint project webpage, http://people.cs.vt.edu/~asandu/Public/GCv7_ADJ
16. Zhu, C., Byrd, R.H., Lu, P., Nocedal, J.: L-BFGS-B: a limited memory FORTRAN code for solving bound constrained optimization problems. Tech. rep., Northwestern University (1994)

Chemical Data Assimilation with CMAQ: Continuous vs. Discrete Advection Adjoints

Tianyi Gou, Kumaresh Singh, and Adrian Sandu

Department of Computer Science, Virginia Tech.,
Blacksburg, VA, 24060, USA
{tygou,kumaresh,sandu}@cs.vt.edu

Abstract. The Community Multiscale Air Quality (CMAQ) system is the Environmental Protection Agency's main modeling tool for atmospheric pollution studies. CMAQ-ADJ, the adjoint model of CMAQ, offers new capabilities such as receptor-oriented sensitivity analysis and chemical data assimilation. This paper presents the construction of discrete advection adjoints in CMAQ. The new adjoints are thoroughly validated against finite differences. We assess the performance of discrete and continuous advection adjoints in CMAQ on sensitivity analysis and 4D-Var data assimilation applications. The results show that discrete adjoint sensitivities better agree with finite difference value than their continuous counterparts. However, continuous adjoints result in a faster convergence of the numerical optimization in 4D-Var data assimilation. Similar conclusions apply to modified discrete adjoints.

Keywords: Chemical Transport Models, Adjoints, Data Assimilation.

1 Introduction

The Community Multiscale Air Quality (CMAQ) modeling system is a powerful third generation air quality modeling tool used to predict the state of the airborne pollutants [1]. CMAQ is capable of modeling important air quality issues such as tropospheric ozone, fine particles, acid deposition, and visibility degradation [2]. CMAQ is the Environmental Protection Agency's main modeling tool for assessing atmospheric pollution, and it is being widely used by the scientific community for a variety of air quality studies.

Adjoint modeling is a powerful tool for computing the sensitivities of the model output with respect to (thereafter, w.r.t.) a large number of model inputs. Adjoints are widely used in applications including sensitivity analysis, data assimilation, parameter estimation, stability analysis, etc. [6]. There are two approaches to adjoint model development. In the *continuous adjoint* approach one first differentiates the underlying mathematical equations at an abstract level, then discretizes the resulting adjoint equations. In the *discrete adjoint* approach one first discretizes the physical equations (using a suitable numerical method), then differentiates the discrete algorithm. Discrete adjoints are popular since they can be obtained by automatic differentiation [7]. The two approaches lead to different adjoint models.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 312–321, 2009.

CMAQ-ADJ is the adjoint model of the CMAQ modeling system, and has been developed through a collaboration between Caltech, Virginia Tech, and University of Houston [2,9,10]. The CMAQ adjoint system has the added capability of performing sensitivity analysis and 4D-Var data assimilation. The current implementation of CMAQ-ADJ considers the advection, diffusion, emission, deposition, and gas-phase processes. The continuous adjoint approach is used for advection processes, while the discrete adjoint approach is used for other processes including chemistry process and diffusion process.

In this paper we present the construction and validation of two newly developed modules CMAQ-ADJ: one is a discrete advection adjoint process, and the second is a modified discrete adjoint, where we remove the monotonic property of the piecewise parabolic method (PPM) [8] used to discretize the horizontal advection process. For the remainder of this paper, we refer to the first one as discrete adjoint version, to the latter one as modified discrete adjoint version, and to the current CMAQ-ADJ advection as the continuous adjoint version. Differences of three adjoint versions for sensitivity analysis and 4D-Var data assimilation are illustrated as well.

This paper is organized as follows. Section 2 introduces the chemical transport model CMAQ. Section 3 presents the construction of discrete and continuous adjoints, as well as their validation. Applications to sensitivity analysis and 4D-Var data assimilation are presented in sections 4 and 5, respectively. Section 6 provides a summary of the work presented in this paper.

2 Chemical Transport Model in CMAQ

Atmospheric Chemical Transport Models (CTMs) are used to capture the knowledge about the physical and chemical processes that govern the evolution of airborne pollutant fields. They have the ability to predict the concentrations of the airborne pollutants. Atmospheric CTMs solve the atmospheric mass balance equation defined as follows [2]:

$$\frac{\partial C_i}{\partial t} = -\mathbf{u} \cdot \nabla C_i + \frac{1}{\rho} \nabla \cdot (\rho \mathbf{K} \nabla C_i) + R_i + E_i \tag{1}$$

where C_i is the concentration for species i, $\mathbf{u}$ is the wind fields, ρ is the air density, $\mathbf{K}$ is the diffusivity tensor, R_i represents the chemical reaction rate for species i and E_i represents emissions. From the above formula, we see that CTMs model the following important processes: advection ($-\mathbf{u} \cdot \nabla C_i$), diffusion $\rho^{-1} \nabla \cdot (\rho \mathbf{K} \nabla C_i)$, chemistry ($R_i$) and emission ($E_i$). In addition to these four parts, CMAQ also includes cloud process, plume-in-grid process and aerosol process. Details of all the science processes can be found in [1,5].

To develop the continuous adjoint system of a CTM we first derive its tangent linear model (TLM) by differentiation, then use Lagrange multipliers and integration by parts to get the adjoint system. Details of deriving adjoint system can be found in [3].

3 Adjoint Construction and Validation

The adjoint of a multi-physics forward CTM requires the construction of adjoints for each of its science processes. As mentioned in the first section, two approaches can be used to develop adjoint model: continuous adjoint approach and discrete adjoint approach.

In this paper we report on a new implementation of a discrete advection adjoint process. Details on adjoint development for other processes can be found in [2].

3.1 Vertical Advection

The vertical advection equation in CMAQ [2] is

$$\frac{\partial(\rho C)}{\partial t} = -\frac{\partial(w\rho C)}{\partial z} \tag{2}$$

where C is concentration vector (a function of time and space), ρ is the air density and w is the vertical wind field.

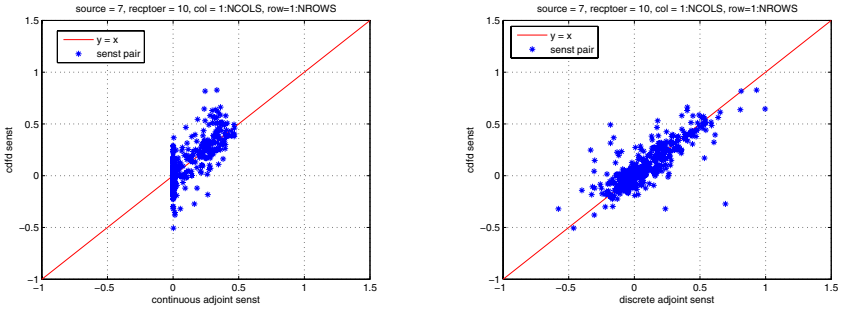
Continuous Adjoint of Vertical Advection. The corresponding continuous adjoint model of vertical advection [2] is

$$-\frac{\partial(\lambda/\rho)}{\partial t} = \frac{\partial(w\lambda/\rho)}{\partial z} \tag{3}$$

where λ is the adjoint variable.

Discrete Adjoint of Vertical Advection. Discrete adjoint of vertical advection is implemented from the forward code with the aid of automatic differential tool such as TAMC (Tangent linear and Adjoint Model Compiler) [7]. For the adjoint mode, we need checkpoints of the forward concentration state. Therefore, we write the concentration into checkpoint files after every dynamic time step in forward mode and read them in the adjoint mode before every dynamic time step.

Validation of Vertical Advection Adjoints. To validate the adjoints of vertical advection, we compare the adjoint sensitivity with central finite difference sensitivity. Specifically, we choose a source layer ($L_s = 7$), a receptor layer ($L_r = 10$), and a species ($S = O_3$, ozone). A perturbation of species S is introduced in each grid of the source layer and is tracked at the corresponding grid (i.e., in the same column) at the receptor layer. In other words, we compute $dC(L_r, S)/dC(L_s, S)$ (for the same column) using both central finite difference approach and the adjoint sensitivity approach. Fig. 1(a) shows the scattered plot of continuous adjoint sensitivity vs. central finite difference values for 8 hours run of CMAQ-ADJ with vertical advection only. Each point represents one sensitivity (one column), with the x coordinate the value obtained by the adjoint method and the y coordinate the value obtained by central differences. Fig. 1(b) shows a similar scattered plot for discrete adjoint sensitivities. We see that discrete adjoint sensitivities of vertical advection agree better with finite difference results than continuous adjoint sensitivities.



(a) Continuous adj. vs. Finite diff. (b) Discrete adj. vs. Finite diff.

Fig. 1. Scattered plot of (a) continuous adjoint sensitivity and (b) discrete adjoint sensitivity vs. finite difference values for an 8 hours run with vertical advection only. Source layer $L_s = 7$, receptor layer $L_r = 10$, and species $S = O_3$ (ozone). Each point represents one sensitivity (one column), with the x coordinate the value obtained by the adjoint method and the y coordinate the value obtained by central differences.

3.2 Horizontal Advection

In CMAQ horizontal advection is directionally split, with successive calls made to x-axis advection and to y-axis advection. Since the same one-dimensional advection algorithm is used for both x and y directions, we consider x-advection in this section. The one-dimensional horizontal advection equation [2] is similar to (2), with the horizontal wind field u replacing w. If the total mass continuity holds for the horizontal advection equation, its equivalent form is

$$\frac{\partial(C)}{\partial t} = -u\frac{\partial(C)}{\partial x} \tag{4}$$

Continuous Adjoint of Horizontal Advection. The corresponding continuous adjoint of horizontal advection takes the form (3) with λ the adjoint variable, ρ the air density, and u (instead of w) the horizontal wind field. The continuous adjoint equation is implemented by calling the forward horizontal advection subroutine with a reversed wind field $(-u)$.

Discrete Adjoint of Horizontal Advection. The same automatic-differentiation based approach used to develop the discrete adjoint of vertical advection is employed to develop the discrete adjoint of horizontal advection.

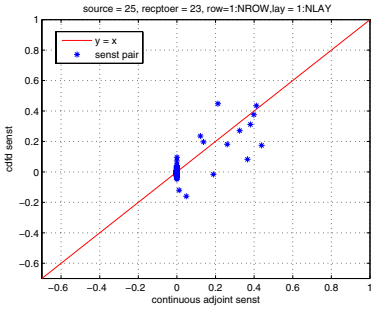
Modified Discrete Adjoint of Horizontal Advection. In CMAQ the numerical method used to solve the advection equation is the piecewise parabolic method (PPM). PPM has monotonicity properties obtained with the help of slope and curvature limiters. The reverse mode differentiation of these limiters may introduce discontinuities in the discrete adjoint solution [8].

In the modified discrete adjoint approach we remove monotonicity characteristics of the PPM by commenting out the steepening procedure (i.e., the limiters);

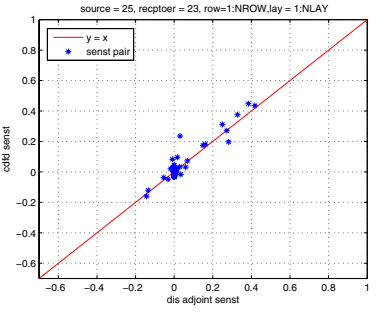
this non-monotonic code is then processed by automatic differentiation. The algorithm is smooth and has no points of non-differentiability.

Validation of Horizontal Advection Adjoints. To validate adjoint of horizontal advection, we plot the adjoint sensitivities and finite difference values . Specifically, we choose a source column (c_s), a receptor column (c_r), and a species ($S = O_3$). For a given vertical layer and all horizontal rows, the sensitivities $dC(c_r, S)/dC(c_s, S)$ are computed using adjoints and using central finite differences.

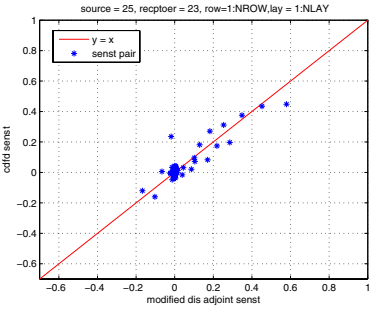
Fig. 2 presents the scattered plots of all three adjoint sensitivities against the finite difference counterparts. An 8-hour horizontal advection experiment is carried out. Each point represents a sensitivity pair at different vertical layer and horizontal row. The continuous adjoint results are shown in Fig. 2 (a), the discrete adjoints in Fig. 2 (b), and the modified discrete adjoints in Fig. 2 (c) shows the scatter plot of the modified discrete adjoint vs. finite difference results. The continuous adjoint sensitivities show the highest discrepancies against the finite differences. Both the discrete adjoint and the modified discrete adjoint results show good agreement with finite difference results.



(a) Continuous adj. vs. Finite diff.



(b) Discrete adj. vs. Finite diff.

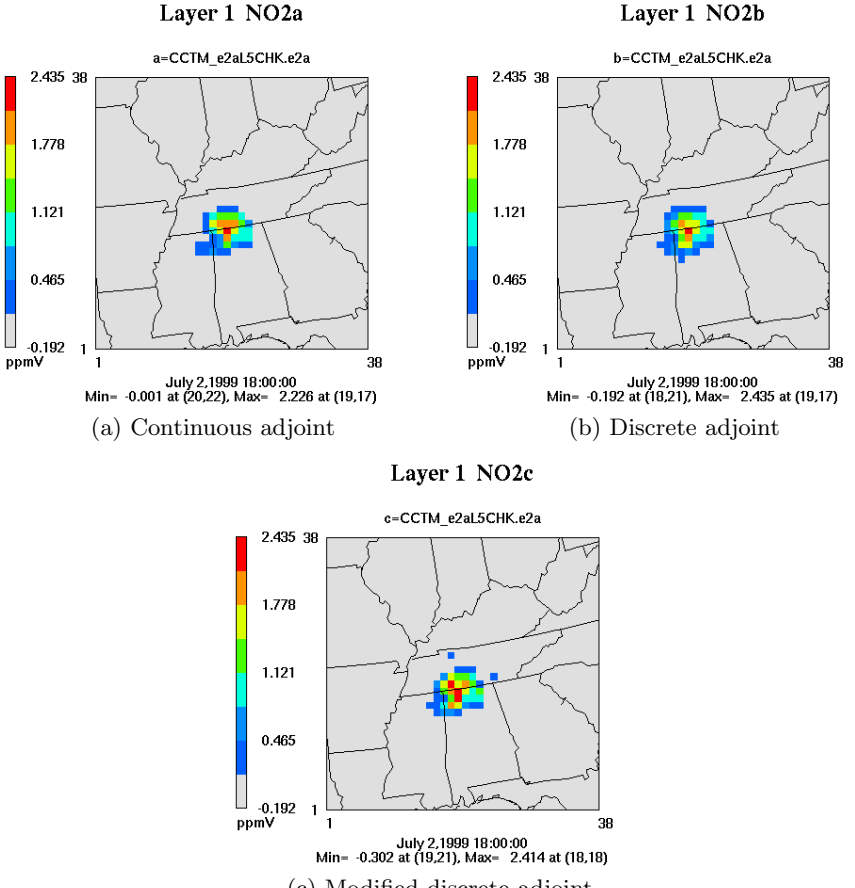


(c) Modified discrete adj. vs. Finite diff.

Fig. 2. Scattered plot of (a) continuous adjoint sensitivity, (b) discrete adjoint sensitivity, and (c) modified discrete adjoint sensitivity vs. finite difference values for 8 hours run of x-horizontal advection only with source column 25, receptor column 23 and species ozone

4 Sensitivity Analysis

Sensitivity analysis quantifies changes in the model output w.r.t. the variations in model inputs. The sensitivity (derivative) of the model output/receptor with respect to model inputs/sources can be obtained by two approaches. One is the forward approach where the perturbation of the input/source propagates forward in time to the output/receptor. The other one is adjoint approach where the perturbation of the receptor is traced backward to the contributing sources. Therefore the forward approach can efficiently compute the sensitivity of many outputs/receptors w.r.t. a few sources. The adjoint approach can efficiently provide the sensitivities of a few outputs/receptors w.r.t. many inputs/sources.



(a) Continuous adjoint

(b) Discrete adjoint

(c) Modified discrete adjoint

Fig. 3. Sensitivity of O_3 at localized receptors at July 3, 1999, 00:00:00 w.r.t. NO_2 source on July 2, 1999, 18:00:00. O_3 receptor location is cols = 18:22, rows = 18:22, layer = 1:4.

With the development of CMAQ-ADJ, CMAQ is capable of performing adjoint sensitivity analysis. To see the difference of sensitivities provided by the three different versions of advection adjoint Fig. 3 presents the results of a 24 hours simulation. The model output is the sum of O_3 concentrations in the area described by grid cells $(18 : 22, 18 : 22, 1 : 4)$, at 00:00:00 (GMT) hours on July 3, 1999. The model inputs are initial NO_2 conditions at 18:00:00 (GMT) hours on July 2, 1999. Fig. 3 reveals small differences between the sensitivity fields computed by the three adjoint versions. We next study how this difference affects the 4D-Var application.

5 4D-Var Data Assimilation

Data assimilation is the process of fusing information from model predictions and observations in order to obtain better initial conditions, boundary conditions or emission estimates. 4D-Var data assimilation optimally combines three sources of information: a priori estimate of the state of the atmosphere; knowledge about the physical and chemical processes that govern the evolution of pollutant fields as captured in the model; and observations of some of the state variables [3].

We apply 4D-Var data assimilation with CMAQ-ADJ to provide optimal estimates of the initial conditions. 4D-Var data assimilation is posed as an optimization problem where the best estimate of initial conditions minimizes the following cost function:

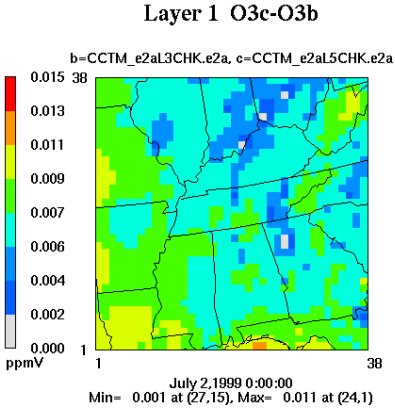
$$J(c^0) = \frac{1}{2}\sum_{k=1}^{N}(c^k - c^{k,obs})^T R_k^{-1}(c^k - c^{k,obs}) + \frac{1}{2}(c^0 - c^b)^T B^{-1}(c^0 - c^b) \quad (5)$$

where c^0 is the initial concentration, c^k is the model prediction at time step k, $c^{(k,obs)}$ is the observation at time step k, R_k is the observation error covariance matrix, c^b is the background concentration and B is the background error covariance matrix. The cost function measures the misfit between model predictions and observations as well as the misfit between initial conditions and background concentrations.

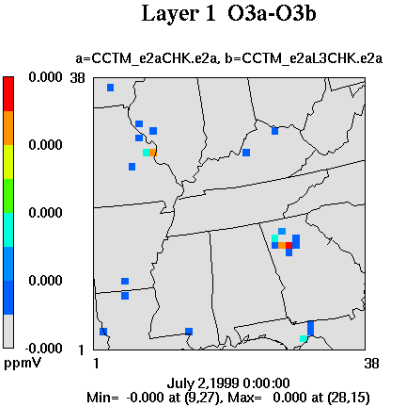
5.1 Implementation

To minimize the cost function (5), we apply L-BFGS, a limited memory quasi-Newton method for solving large scale optimization problems [4]. L-BFGS is an iterative method which requires, at each iteration step, the cost function value and its gradient at that point, and returns another point which is a better approximation to the optimal solution. The whole process continues until the convergence criteria are satisfied.

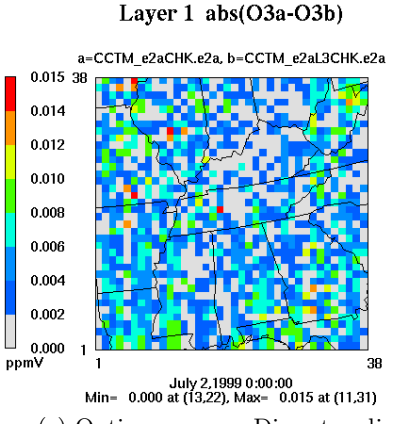
In the implementation of CMAQ/4D-Var we interface L-BFGS with the CMAQ-ADJ. The cost function value and gradient value are obtained through a forward run followed by an adjoint run of CMAQ-ADJ.



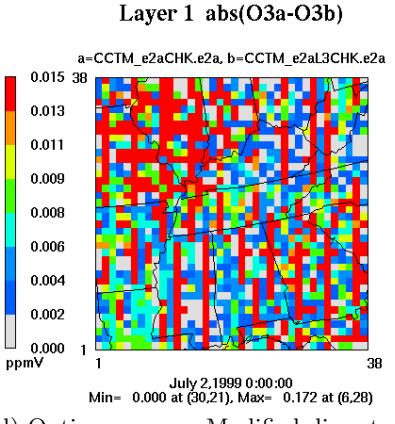
(a) Perturbation in initial conditions.



(b) Optimum error: Continuous adj.



(c) Optimum error: Discrete adj.

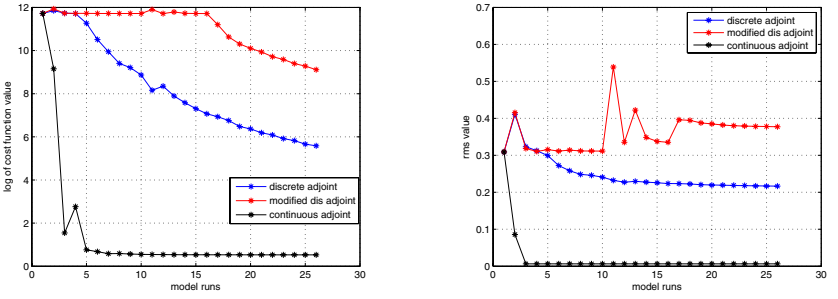


(d) Optimum error: Modified discrete adj.

Fig. 4. 4D-Var data assimilation carried out for 12 hours to optimize the initial O_3 values. (a) difference between perturbed and reference concentration at initial time; (b) - (d) difference between optimized and reference concentration at initial time using (b) continuous adjoint; (c) discrete adjoint; (d) modified discrete adjoint.

5.2 4D-Var Results

To validate our CMAQ/4D-Var implementation we have designed a test case as follows. First, we choose a reference initial concentration c_0 and perform a forward model run to lay down a set of synthetic observations. Second, a perturbation is introduced to obtain $c_p = c_0 + \Delta c_0$. Third, $c^b = c_p$ is used as the background value and as the starting point for L-BFGS algorithm. After each iteration run, we obtain a new initial point c_0^k. Again, by performing a forward run and a adjoint run, we obtain the cost function value and gradient at c_0^k, which are used as the input for the next L-BFGS iteration. When convergence occurs we get an optimal estimate c_0^a to the reference initial condition c_0.



(a) Cost function (b) RMS values

Fig. 5. Convergence of L-BFGS optimization. (a) Log of the cost function values vs. model runs; (b) RMS values vs. model runs.

In the following section, we show the 4D-Var simulation results for three different versions of CMAQ-ADJ. In our experiment, we perturbed ozone and retrieved ozone only. We visualize the difference between the initial guess and the reference as well as the difference between the optimized solution and the reference. In addition, we plot the log of the cost function values as the iteration number increases. To see if the optimized solution converges to the reference, we also plot the root-mean square (RMS) values given by: RMS= $\|c_0^k - c_0\| \, / \|c_0\|$.

Fig.4 and Fig.5 show the 4D-Var validation results for a 12 hours simulation with all science processes for species O_3. With the continuous adjoint the optimized initial condition agrees very well with the reference initial condition. In addition, the cost function and RMS values decease significantly with the number of model runs indicating a fast convergence to the reference initial condition. On the other hand, the discrete adjoint shows slow convergence, and the modified discrete adjoint does not achieve convergence.

6 Conclusion

In this paper we present the construction and validation of discrete and modified discrete advection adjoints in CMAQ. Discrete adjoint sensitivities given by both approaches match the finite difference results better than continuous adjoint sensitivities. However, in 4D-Var data assimilation experiments, a much faster convergence is obtained when the gradients are computed using the continuous adjoint approach than using the discrete adjoint or modified adjoint approaches. We believe that this effect is related to the smoothness of the respective gradients. Accordingly, we conclude that, for CMAQ-ADJ, continuous adjoint is preferable for 4D-Var data assimilation while discrete adjoint provide better sensitivity agreement with finite difference results. Both continuous and discrete adjoint implementations are available to the user in the new CMAQ-ADJ version.

References

1. Byun, D.W., Ching, J.K.S.: Science Algorithms of the EPAModels-3 Community Multiscale Air Quality (CMAQ) Modeling System; U.S. EPA/600/R-99/030; U.S. Environmental Protection, Agency: Research Triangle Park, NC (1999)
2. Hakami, A., Henze, D.K., Seinfeld, J.H., Singh, K., Sandu, A., Kim, S., Byun, D., Li, Q.: The Adjoint of CMAQ. Environ. Sci. Technol. 41(22), 7807–7817 (2007)
3. Sandu, A., Daescu, D.N., Carmichael, G.R., Chai, T.F.: Adjoint sensitivity analysis of regional air quality models. J. Comput. Phys. 204, 222–252 (2005)
4. Liu, D.C., Nocedal, J.: On the limited memory BFGS method for large-scale optimization. Math. Programming 45, 503–528 (1989)
5. Byun, D.W., Schere, K.: Review of the Governing Equations, Computational Algorithms, and Other Components of the Models-3 Community Multiscale Air Quality (CMAQ) Modeling System. Applied Mechanics Reviews 59, 51–77 (2006)
6. Errico, R.M.: What is an adjoint model? Bull. Amer. Meteor. Soc. 78, 2577–2591 (1997)
7. Giering, R., Kaminski, T.: Recipes for Adjoint code Construction. ACM Trans. Math. Softw. 24, 437–474 (1998)
8. Colella, P., Woodward, P.R.: The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations. J. Comput. Phys. 54, 174–201 (1984)
9. Singh, K., Sandu, A., Hakami, A. and Seinfeld: CMAQ-ADJv4.5 (2007), `http://people.cs.vt.edu/~asandu/Software/CMAQ_ADJ/CMAQ_ADJ.html`
10. Singh, K., Sandu, A., Hakami, A., Seinfeld, J.: CMAQ v4.5 Adjoint Users's Manual (2007)

A Second Order Adjoint Method to Targeted Observations

Humberto C. Godinez and Dacian N. Daescu

Department of Mathematics and Statistics,
Portland State University, Portland OR 97207
{hgodinez,daescu}@pdx.edu

Abstract. The role of the second order adjoint in targeting strategies is studied and analyzed. Most targeting strategies use the first order adjoint to identify regions where additional information is of potential benefit to a data assimilation system. The first order adjoint posses a restriction on the targeting time for which the linear approximation accurately tracks the evolution of perturbation. Using second order adjoint information it is possible to maintain some accuracy for longer time intervals, which can lead to an increase on the target time. We propose the use of the dominant eigenvectors of the Hessian matrix as an indicator of the directions of maximal error growth for a given targeting functional. These vectors are a natural choice to be included in the targeting strategies given their mathematical properties.

1 Introduction

Observation targeting strategies aim to identify optimal regions where supplemental data can improve the forecast of a data assimilation system. Adjoint modeling has been an essential tool for the development of targeting strategies in the context of variational data assimilation methods. The adjoint of the tangent linear model associated to an atmospheric model is a key ingredient to implementing various targeting strategies, such as gradient sensitivity, dominant singular vectors, and sensitivity to observations ([1], [2], [3], [4], [5], [6]).

The first order adjoint (FOA) model provides the gradient of a scalar-valued forecast aspect, typically a forecast error measure. As such, the FOA represents a first order approximation to the evolution of perturbations in the atmospheric model. The accuracy of this approximation is limited by the magnitude of the perturbation and by the time length of the forecast. As the forecast time lead increases, the accuracy of the FOA to track the initial-condition error propagation is impaired. This poses a limitation on the time window for which targeting strategies based on the FOA fields are reliable. To overcome this practical difficulty and to increase the effectiveness of adjoint targeting strategies, a second order adjoint (SOA) model may be considered to capture the quadratic terms in the error growth approximation. An overview of the SOA model implementation and applications to variational data assimilation is provided in [7].

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 322–331, 2009.

In our work a targeting strategy based on SOA modeling is considered and numerical experiments are presented in a comparative analysis between the first order and the second order adjoint-based observation targeting guidance. The importance of incorporating SOA information is investigated by using first and second order Taylor approximations to model the nonlinear error growth and perturbations in a forecast error functional.

Section 2 briefly revisits the four dimensional variational (4D-Var) data assimilation and the FOA and SOA models. In section 3 the implementation of the FOA and SOA models to a shallow water (SW) model is presented. First and second order Taylor approximations to the perturbations in a forecast error functional are analyzed. A novel targeting strategy based on the eigenvalues and eigenvectors of the Hessian matrix of the forecast aspect is implemented in section 4 using the SW model. Conclusions and future work are in section 5.

2 Data Assimilation and Adjoint Modeling

Given an initial state $\mathbf{x}_0$, let $\mathcal{M}_i$ denote the discrete atmospheric model (forward model) that evolves the state from t_i to t_{i+1}

$$\mathbf{x}_{i+1} = \mathcal{M}_i\left(\mathbf{x}_i\right), \quad i = 0, \ldots, N-1. \tag{1}$$

Data assimilation techniques [8] combine information from a dynamical model, a prior (background) estimate, and observational data to provide an optimal initial condition (analysis) to the dynamical system (1). The 4D-Var analysis [9] is obtained by minimizing a cost functional that measures the discrepancy between the model state, background estimate, and time distributed observational data

$$\begin{aligned}\mathcal{J}\left(\mathbf{x}_0\right) = \left(\mathbf{x}_0 - \mathbf{x}^b\right)^T \mathbf{B}^{-1} \left(\mathbf{x}_0 - \mathbf{x}^b\right) + \\ \sum_{i=0}^{k} \left(\mathbf{y}_i - H_i\left[\mathbf{x}_i\right]\right)^T \mathbf{R}_i^{-1} \left(\mathbf{y}_i - H_i\left[\mathbf{x}_i\right]\right)\end{aligned} \tag{2}$$

where $\mathbf{x}^b$ is the background, $\mathbf{y}_i$ is the observation vector at t_i, $\mathbf{B}$ and $\mathbf{R}$ are the error covariance matrices for the background and observations, respectively, and H_i is the observational operator mapping the state into observations at t_i.

Adaptive observations are supplementary data collected to reduce the error of some aspect of the forecast at verification time $t_v > t_k$ over a verification domain $\mathcal{D}_v$, expressed as

$$\mathcal{J}_v\left(\mathbf{x}_v\right) = \frac{1}{2} \left\langle \mathbf{P}\left(\mathbf{x}_v - \mathbf{x}_v^t\right), \mathbf{P}\left(\mathbf{x}_v - \mathbf{x}_v^t\right) \right\rangle_{\mathbf{E}} \tag{3}$$

where $\mathbf{x}_v^t$ is the true state at the verification time, $\mathbf{x}_v$ is the state of the system at time t_v, $\mathbf{P}$ is a projection operator on $\mathcal{D}_v$ satisfying $\mathbf{P}^*\mathbf{P} = \mathbf{P}^2 = \mathbf{P}$. The inner product $\langle , \rangle_{\mathbf{E}}$ is defined as $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{E}} = \langle \mathbf{x}, \mathbf{E}\mathbf{y} \rangle$, where $\mathbf{E}$ is a symmetric positive definite matrix, typically chosen to induce the total energy norm. The measure (3) is the *forecast error functional* at the verification time t_v over the verification domain $\mathcal{D}_v$.

2.1 Taylor Expansion of the Forecast Error Functional

The functional $\mathcal{J}_v$ implicitly depends on the initial condition $\mathbf{x}_0$ of (1)

$$\mathcal{J}_v\left(\mathbf{x}_N\right) = \mathcal{J}_v\left(\mathcal{M}_{t_0 \to t_N}\left(\mathbf{x}_0\right)\right), \tag{4}$$

where $\mathcal{M}_{t_0 \to t_N}$ is the nonlinear model integration from t_0 to $t_N = t_v$,

$$\mathcal{M}_{t_0 \to t_N} = \mathcal{M}_{N-1} \circ \cdots \circ \mathcal{M}_0\left(\mathbf{x}_0\right). \tag{5}$$

A perturbation $\delta\mathbf{x}_0$ in the initial condition will result in a perturbation $\delta\mathcal{J}_v\left(\mathbf{x}_N\right) = \mathcal{J}_v\left(\mathbf{x}_N + \delta\mathbf{x}_N\right) - \mathcal{J}_v\left(\mathbf{x}_N\right)$ that, to a second order Taylor approximation, can be expressed

$$\delta\mathcal{J}_v\left(\mathbf{x}_N\right) \approx \nabla_{\mathbf{x}_0}\mathcal{J}_v\left(\mathbf{x}_N\right)\delta\mathbf{x}_0 + \frac{1}{2}\delta\mathbf{x}_0^T\nabla^2_{\mathbf{x}_0}\mathcal{J}_v\left(\mathbf{x}_N\right)\delta\mathbf{x}_0 \tag{6}$$

The gradient $\nabla_{\mathbf{x}_0}\mathcal{J}_v\left(\mathbf{x}_N\right)$ is obtained through the FOA model associated to (1)

$$\lambda_N = \nabla_{\mathbf{x}_N}\mathcal{J}_v\left(\mathbf{x}_N\right) \tag{7}$$

$$\lambda_i = \mathbf{M}_i^*\left(\mathbf{x}_i\right)\lambda_{i+1}, \quad i = N-1, \ldots, 0 \tag{8}$$

where $\mathbf{M}_i$ is the derivative (tangent linear model) of $\mathcal{M}_i$, and $\mathbf{M}_i^*$ its adjoint.

Second order derivative information, as the product of the Hessian $\nabla^2_{\mathbf{x}_0}\mathcal{J}_v\left(\mathbf{x}_N\right)$ times a user-defined vector, may be obtained by integration of a SOA model.

2.2 SOA Model Equations

The equations of the discrete SOA model associated to (1) and (4) are

$$\nu_N = \nabla^2_{\mathbf{x}_N}\mathcal{J}_v\left(\mathbf{x}_N\right)\mu_N = \mathbf{P}^T\mathbf{E}\mathbf{P}\mu_N \tag{9}$$

$$\nu_i = \mathbf{M}_i^*\left(\mathbf{x}_i\right)\nu_{i+1} + \frac{\partial}{\partial\mathbf{x}_i}\left[\mathbf{M}_i^*\left(\mathbf{x}_i\right)\bar{\lambda}_{i+1}\right]\mu_i, \quad i = N-1, \ldots, 0 \tag{10}$$

where μ is the solution to the tangent linear model (TLM)

$$\mu_0 = \mathbf{w} \tag{11}$$

$$\mu_{i+1} = \mathbf{M}_i\left(\mathbf{x}_i\right)\mu_i, \quad i = 0, \ldots, N-1, \tag{12}$$

$\mathbf{w}$ is an user-defined vector, and the notation $\bar{\lambda}_{i+1}$ in the last term of (10) indicates that the state derivative applies to the $\mathbf{M}_i^*\left(\mathbf{x}_i\right)$ operator only while treating the adjoint variables λ_{i+1} as constants ([10], [7]).

The solution of the SOA model (9)-(10) provides the Hessian vector product $\nabla^2_{\mathbf{x}_0}\mathcal{J}\left(\mathbf{x}_0\right)\mu_0 = \nu\left(t_0\right)$ that is required to evaluate the second order term in the Taylor approximation (6), thus providing the quadratic term for the evolution of perturbations in the forward model.

3 The SW Model, FOA and SOA Taylor Approximations

A global 2D shallow water (SW) model on a sphere is used for the numerical experiments. The model describes the hydrodynamic flow on a sphere under the assumptions that the vertical motion is much smaller that the horizontal motion. It is also assumed that the fluid depth is small compared with the radius of the sphere (radius of Earth). The equations of the SW model are

$$\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = -f\mathbf{k}\times\mathbf{v} - \nabla\phi, \tag{13}$$

$$\frac{\partial\phi}{\partial t} = -\nabla\cdot\left[(\phi-\phi_s)\,\mathbf{v}\right], \tag{14}$$

where $\dfrac{\mathrm{d}}{\mathrm{d}t} = \dfrac{\partial}{\partial t} + \mathbf{v}\cdot\nabla$, $\mathbf{v} = u\mathbf{i} + v\mathbf{j}$ with $\mathbf{i},\mathbf{j},\mathbf{k}$ being the unit vectors in the three orthonormal directions on the sphere, u and v are the zonal and meridional velocity components, respectively, h is the fluid depth, h_s is the bottom topography, g the gravitational constant, $\phi = gh$, $\phi_s = gh_s$, and f is the Coriolis parameter. The norm used on the state space $\mathbf{x}$ is the total energy norm, induced by the inner product

$$\langle \mathbf{x},\mathbf{x}\rangle_{\mathbf{E}} = \frac{1}{2}\left(u^2+v^2\right) + \frac{g}{h_0}h^2.$$

A Godunov type finite volume discretization method is used to discretize the SW equations. The Van-Leer transport scheme, as described in [11], is employed for the space discretization. Computations are done on a $2.5^\circ \times 2.5^\circ$ grid with a time step $\Delta t = 450$s and the verification time is set at $t_v = t_0 + 24$h. The reference state ('truth') $\mathbf{x}_0^t$ is taken from the trajectory produced by a numerical integration of the SW model using as initial condition the 500hPa ERA-40 data set from the European Centre for Medium-Range Weather Forecasts (ECMWF), valid for March 15 2002 at 06 : 00 hours. The background state $\mathbf{x}^b$ is taken from a 6-hour model simulation initialized at $t_0 - 6$h with the ERA-40 data set valid for March 15 2002 at 00 : 00 hours. The difference between the 24h forecasts initiated from $\mathbf{x}_0^t$ and $\mathbf{x}^b$, respectively, exhibits a high discrepancy in the region $[55^\circ W, 35^\circ W]\times[52^\circ N, 65^\circ N]$ which is taken as the verification domain $\mathcal{D}_v$ at t_v.

The discrete TLM, FOA, and SOA models are obtained using the Automatic Differentiation package TAMC [12]. The discrete SOA can be seen as the action of the Hessian matrix of the scalar forecast aspect of interest on a vector. The code for the second order adjoint can be computed in the forward over reverse mode, this is, taking the tangent of a forward-backward integration.

3.1 Taylor Approximation with Adjoint Models

The Taylor approximation (6) is valid for relatively small perturbations of the initial condition $\mathbf{x}_0$ and a short forecast time lead, depending on the nonlinearity of the forward model. Taking the SW model as the forward model, forecast perturbations are computed together with their first and second order Taylor

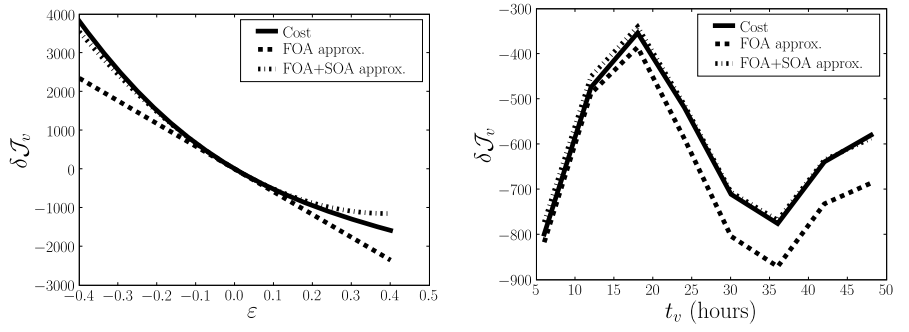


Fig. 1. Left figure: Taylor approximation (6) of the perturbation of the forecast error functional (3) as a function of initial condition perturbations coefficient ε. Right figure: Time evolution of (6) as a function of the verification time t_v, with $\varepsilon = 0.1$.

approximations using the FOA and SOA models. To corroborate the accuracy of the approximations to $\delta\mathcal{J}_v$, the initial condition, taken from the background state, is perturbed according to

$$\mathbf{x}_0(\varepsilon) = \mathbf{x}_0 + \delta\mathbf{x}_0(\varepsilon), \quad \delta\mathbf{x}_0(\varepsilon) = \varepsilon\left(\mathbf{x}_0 - \mathbf{x}^t(t_0)\right)$$

where ε is a coefficient that controls the perturbation in the initial condition of the forward model. The perturbation $\delta\mathcal{J}_v$, as well as the adjoint-based approximations, are computed for values of the perturbation coefficient ε ranging from -0.4 to 0.4 with increments of 0.01 and fixed $t_v = 24$h, then for a time-varying forecast lead $t_v - t_0$ ranging from 6-hour to 72-hour with one hour increments.

Figure 1 (left) shows the perturbation $\delta\mathcal{J}_v$ of the forecast error, and its first and second order Taylor approximations. It is noticed that the second order approximation remains accurate over a wide range of perturbations as compared to the first order approximation.

Figure 1 (right) shows the time evolution of the Taylor approximation using the FOA and SOA models, as a function of t_v, while keeping $\varepsilon = 0.1$ fixed. As the verification time increases the second order approximation remains significantly more accurate than the first order approximation. Similar results were obtained with various values of $\varepsilon \neq 0$.

It must be noticed that the perturbation growth is time dependent, this is, the solutions of the forward and adjoint models depend on the verification time t_v. An important question to address is the accuracy of the approximation as the verification time is increased. The approximation can lose accuracy if there is a strong nonlinear time dependence of the forward model which is not accurately captured in the adjoint models. Traditional targeting strategies account only for linear error propagation and differ on the selection of the norm used to measure the error growth propagation (e.g. total energy vs. error covariance metric [3]). The limited accuracy of the first order approximation is a major difficulty in

extending the targeting time interval and the use of second order derivative information may prove to be of relevance in practical applications.

4 Targeting Using SOA Information

Applications of the FOA sensitivity analysis during field experiments to collect targeted observations are presented in [1], which gives the fundamentals for targeting strategies based on adjoint modeling.

The FOA model provides the gradient of the forecast error functional $\mathcal{J}_v$ with respect to the initial condition $\mathbf{x}_0$ of the forward model. The gradient is used to define a space-distributed sensitivity field

$$F_v = \|\nabla_{\mathbf{x}_0} \mathcal{J}_v\|_2 \tag{15}$$

where the 2-norm is taken at each grid-point on the mesh. In the FOA targeting approach supplementary observations are taken at locations where F_v exhibits the largest magnitude.

To accurately track the propagation of perturbations, we propose a new targeting method to incorporate SOA information.

Consider the second term in the Taylor approximation (6)

$$\frac{1}{2}\delta\mathbf{x}_0^T \mathbf{H} \delta\mathbf{x}_0, \tag{16}$$

where $\mathbf{H} = \nabla^2_{\mathbf{x}_0} \mathcal{J}_v(\mathbf{x}_N)$ is the Hessian matrix of $\mathcal{J}_v$. Without loss of generality, consider initial perturbations with unit two-norm, that is $\|\delta\mathbf{x}_0\|_2 = 1$, then (16) is a *Rayleigh-Ritz* ratio

$$\frac{\delta\mathbf{x}_0^T \mathbf{H} \delta\mathbf{x}_0}{\delta\mathbf{x}_0^T \delta\mathbf{x}_0}. \tag{17}$$

The vector for which the Rayleigh-Ritz ratio (17) is maximized provides the direction of maximal quadratic error propagation in the second order term of the Taylor approximation. Since $\mathbf{H}$ is a symmetric matrix, the maximum of (17) is provided by the eigenvector associated with the leading eigenvalue of $\mathbf{H}$ [13].

In [4], [3] the singular vectors of the tangent linear model are used to define a sensitivity field for targeted observations. Following a similar approach, we define the sensitivity function based on the leading eigenvectors of the Hessian.

Let σ_i be the ith eigenvalue of $\mathbf{H}$, ordered so that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$, and let $\mathbf{v}_i$ be its corresponding eigenvector. Consider the first m leading eigenvectors $\mathbf{v}_i, i = 1, \ldots, m$, where $m << n$, n being the dimension of the matrix. The SOA-based sensitivity field is defined as

$$F_m = \sum_{i=1}^{m} \frac{\sigma_i}{\sigma_1} \|\mathbf{v}_i\|_2^2, \tag{18}$$

were the norm of the eigenvector is evaluated on each grid-point of the mesh.

4.1 Eigenvectors of the Hessian of the SW Forecast

At the 2.5-degree grid resolution the dimension of the discrete state vector $\mathbf{x}$ is $n \sim 3 \times 10^4$, such that the number of entries in the Hessian matrix $\mathbf{H}$ is of order of 10^9. Storing such a matrix is clearly unpractical, however, iterative methods that require the action of the matrix on a vector can be implemented to obtain the Hessian eigenvalues and eigenvectors. The Arnoldi Package (ARPACK) [14] is used to compute the leading eigenpairs of $\mathbf{H}$. For one matrix vector product of the Hessian the tangent linear model is integrated forward in time to the verification time t_v and the FOA and SOA models are integrated backward in time to any targeting instance t_i of the data assimilation window.

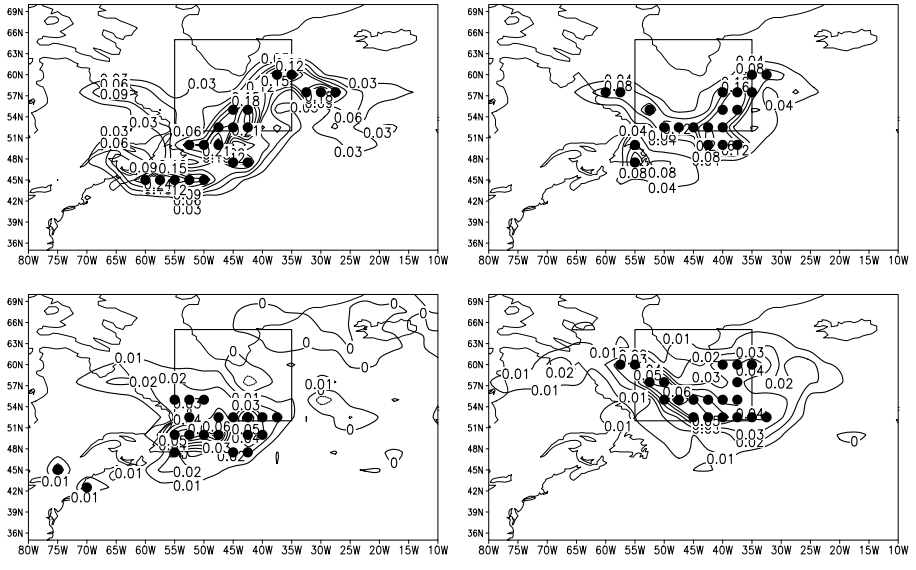


Fig. 2. Top figures: sensitivity field (15) at $t_i = t_0$ (left) and at $t_i = t_0 + 6$h (right). Bottom figures: sensitivity field (18) with $m = 10$ eigenvectors at $t_i = t_0$ (left) and at $t_i = t_0 + 6$h (right). The verification domain $\mathcal{D}_v$ is the region within the rectangle.

The setup for the reference and control simulations, as well as the verification domain and verification time, are the same as in section 3. For comparison, the sensitivity fields obtained with the FOA sensitivity function (15) are shown in figure 2 for $t_i = t_0$ (top-left) and $t_i = t_0 + 6$h (top-right). The locations of adaptive observations (marked with •) correspond to the grid points where the sensitivity fields have the largest magnitude. Figure 2 also shows the sensitivity field obtained with (18), using $m = 10$ leading eigenvectors of the Hessian matrix, at $t_i = t_0$ (bottom-left) and $t_i = t_0 + 6$h (bottom-right). The difference in the sensitivity fields between the FOA and SOA methods illustrates the different type of perturbation growth being measured. Both FOA and SOA fields are time-varying, thus the location of targeted observations depends on the targeting instant.

Table 1. Leading eigenvalues of the Hessian matrix for $t_i = 0$h to 6h, at 1-hour increments and corresponding CPU time for the computation of the leading 10 eigenvalues and eigenvectors at each time instance

t_i (h)	0	1	2	3	4	5	6
σ_1	8.455	7.272	6.721	5.932	5.270	4.355	3.541
CPU (s)	1698.52	1651.28	1589.41	1542.74	1477.11	1429.97	1368.50

Table 1 shows the leading eigenvalue of $\mathbf{H}$ for a targeting time t_i from 0h to 6h, at 1-hour increments. The leading eigenvalue decreases in magnitude as t_i increases, which may indicate that as the forecast time lead is shortened the impact of the second order derivative information decreases. This is consistent with the analysis in section 3.1, where the contribution of the SOA was shown to diminish for forecasts closer to the initial time. Table 1 also shows the CPU time, in seconds, for the computation of 10 leading eigenpairs of the Hessian matrix at each hour on a 1.86GHz Xeon Quad Core 5320 Processor. The computational overhead of a SOA integration is about 3 times that of a FOA integration.

4.2 Targeted Observations and Data Assimilation Experiments

To illustrate the SOA targeting strategy we apply a 4D-Var data assimilation scheme with adaptive observations to the SW model. The assimilation window is $[0\text{h}, 6\text{h}]$ with the verification time set $t_v = 24$h. A first assimilation experiment is performed with 20 adaptive observations placed at $t = 0$h where the sensitivity field (18) has the highest values, as marked in figure 2 (bottom-left). The performance of adaptive observations obtained from the eigenvector sensitivity function (18) is compared with that obtained from the FOA sensitivity function (15), as marked in figure 2 (top-left).

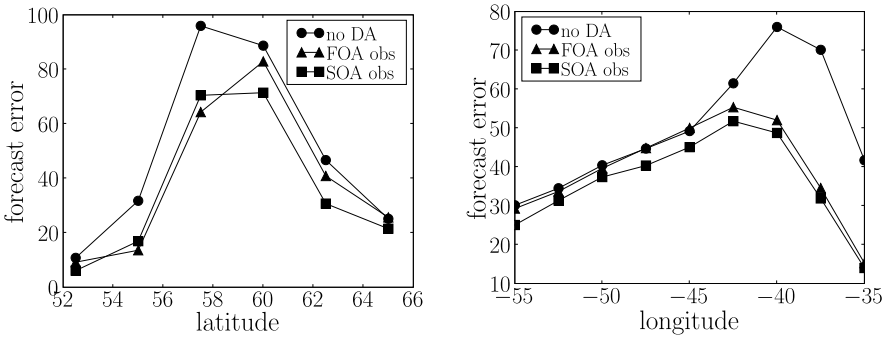


Fig. 3. Longitudinal (left) and latitudinal (right) forecast error average over the verification domain without data assimilation (circles), with data assimilation using adaptive observations from FOA (triangles), and adaptive observations from eigenvectors of the Hessian (squares)

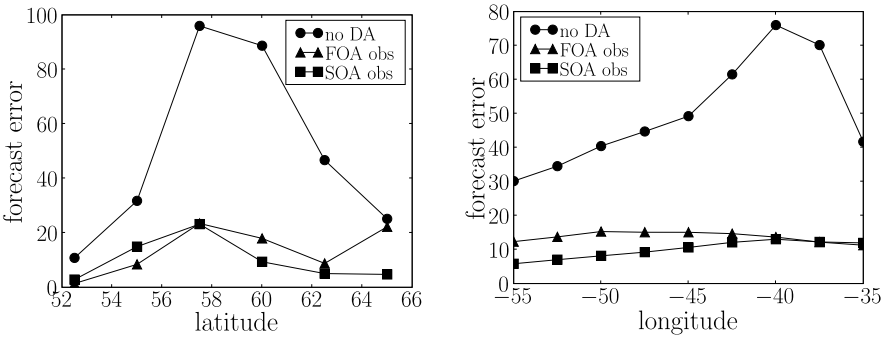


Fig. 4. Longitudinal (left) and latitudinal (right) forecast error average over the target domain without data assimilation (circles), with data assimilation using adaptive observations at both $t = 0$h and $t = 6$h from FOA (triangles), and eigenvectors of the Hessian (squares)

Figure 3 shows the longitudinal (left) and latitudinal (right) forecast error average over the verification domain. It is noticed that the SOA targeting guidance improved the overall forecast quality, as compared to the FOA methodology.

In a second experiment, 20 adaptive observations are placed at both $t = 0$h and 6h. Figure 4 shows the longitudinal and latitudinal averages of the forecast error, where again it is noticed an improvement in the forecast quality from the adaptive observations obtained with the SOA field (eigenvectors) over the adaptive observations obtained with the FOA sensitivity field. It is also noticed that insertion of targeted observations at $t = 6$h is of significant benefit to the forecast, indicating a larger forecast impact from these observations. This is consistent to the observation sensitivity study in [6] where is was found that the forecast sensitivity to observations increases for observations near the end of the assimilation window, and thus closer to the verification time. In addition, accounting for data interaction is essential when multiple targeting instants are considered [15], and this is an area where further research is much needed. Nevertheless, the results show the relevance of the SOA in targeting strategies when dealing with large time intervals and/or highly nonlinear forecast models.

5 Conclusions

Properly accounting for nonlinear error growth is an unresolved issue in targeted observations for numerical weather prediction. In this study a novel approach based on second order adjoint modeling is proposed to account for the quadratic initial-condition error growth in the model forecast. Preliminary numerical experiments indicate that the SOA methodology is effective and may outperform the traditional FOA approach to observation targeting. Further experiments are required to validate this approach in realistic models. To fully exploit the benefits of the SOA model, novel targeting strategies must combine information from both FOA and SOA models to form a more cohesive and accurate strategy.

Acknowledgments. This research was supported by the NASA Modeling, Analysis and Prediction Program under award NNG06GC67G and by the 2006 Intel Oregon Faculty Fellowship program. The work of the first author was also supported by a scholarship from Consejo Nacional de Ciencia y Technologìa.

References

1. Langland, R.H., Gelaro, R., Rohaly, G.D., Shapiro, M.A.: Targeted observations in FASTEX: Adjoint-based targeting procedures and data impact experiments in IOP17 and IOP18. Q. J. R. Meteorol. Soc. 125, 3241–3270 (1999)
2. Langland, R.H.: Issues in targeted observing. Q. J. R. Meteorol. Soc. 131, 3409–3425 (2005)
3. Palmer, T.N., Gelaro, R., Barkmeijer, J., Buizza, R.: Singular Vectors, Metrics, and Adaptive Observations. J. Atmos. Sci. 55, 633–653 (1998)
4. Buizza, R., Montani, A.: Targeting Observations Using Singular Vectors. J. Atmos. Sci. 56, 2965–2985 (1999)
5. Baker, N.L., Daley, R.: Observation and background adjoint sensitivity in the adaptive observation-targeting problem. Q. J. R. Meteorol. Soc. 126, 1431–1454 (2000)
6. Daescu, D.N.: On the Sensitivity Equations of Four-Dimensional Variational (4D-Var) Data Assimilation. Mon. Wea. Rev. 136, 3050–3065 (2008)
7. Le Dimet, F.X., Navon, I.M., Daescu, D.N.: Second-Order Information in Data Assimilation. Mon. Wea. Rev. 130, 629–648 (2002)
8. Kalnay, E.: Atmospheric Modeling, Data Assimilation, and Predictability, p. 364. Cambridge University Press, Cambridge (2002)
9. Le Dimet, F.X., Talagrand, O.: Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. Tellus 38A, 97–110 (1986)
10. Daescu, D.N., Navon, I.M.: Efficiency of a POD-based reduced second-order adjoint model in 4D-Var data assimilation. Int. J. Numer. Meth. Fluids 53, 985–1004 (2007)
11. Lin, S.J., Chao, W.C., Sud, Y.C., Walker, G.K.: A Class of the van Leer-type Transport Schemes and Its Application to the Moisture Transport in a General Circulation Model. Mon. Wea. Rev. 122, 1575–1593 (1994)
12. Giering, R.: Tangent linear and Adjoint Model Compiler, Users manual. Center for Global Change Sciences, Department of Earth, Atmospheric, and Planetary Science. MIT, Cambridge (1997)
13. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1985)
14. Lehoucq, R.B., Sorensen, D.C., Yang, C.: ARPACK Users' Guide: Solution of Large-scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods. Society for Industrial and Applied Mathematics, Philadelphia (1998)
15. Daescu, D.N., Navon, I.M.: Adaptive observations in the context of 4D-Var data assimilation. Meteorol. Atmos. Phys. 85, 205–226 (2004)

A Scalable and Adaptable Solution Framework within Components of the Community Climate System Model

Katherine J. Evans[1], Damian W.I. Rouson[2], Andrew G. Salinger[3], Mark A. Taylor[3], Wilbert Weijer[4], and James B. White III[1]

[1] Oak Ridge National Laboratory, Oak Ridge, TN 37831
[2] Sandia National Laboratory, Livermore, CA 94551
[3] Sandia National Laboratory, Albuquerque, NM 87185
[4] Los Alamos National Laboratory, Los Alamos, NM 87545

Abstract. A framework for a fully implicit solution method is implemented into (1) the High Order Methods Modeling Environment (HOMME), which is a spectral element dynamical core option in the Community Atmosphere Model (CAM), and (2) the Parallel Ocean Program (POP) model of the global ocean. Both of these models are components of the Community Climate System Model (CCSM). HOMME is a development version of CAM and provides a scalable alternative when run with an explicit time integrator. However, it suffers the typical time step size limit to maintain stability. POP uses a time-split semi-implicit time integrator that allows larger time steps but less accuracy when used with scale interacting physics. A fully implicit solution framework allows larger time step sizes and additional climate analysis capability such as model steady state and spin-up efficiency gains without a loss in scalability. This framework is implemented into HOMME and POP using a new Fortran interface to the Trilinos solver library, ForTrilinos, which leverages several new capabilities in the current Fortran standard to maximize robustness and speed. The ForTrilinos solution template was also designed for interchangeability; other solution methods and capability improvements can be more easily implemented into the models as they are developed without severely interacting with the code structure. The utility of this approach is illustrated with a test case for each of the climate component models.

1 Introduction

Climate simulation will not grow to the ultrascale without new algorithms to overcome the scalability barriers blocking existing implementations. Until recently, climate simulations concentrated on the question of whether the climate is changing. The emphasis is now shifting to impact assessments, mitigation and adaptation strategies, and regional details. Such studies will require significant increases in spatial resolution and model complexity while maintaining adequate throughput. The barrier to progress is the resulting decrease in time step without increasing single-process performance [1].

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 332–341, 2009.

To be able to run higher resolution global climate models, several different approaches have been taken to minimize computation time. For dynamical cores with good scalability, the whole system is solved explicitly, and increasing processor counts are utilized as grids are refined [2]. More commonly, the time integration of the climate system is integrated semi-implicitly; the relatively slower physics is solved explicitly with a larger time step size and the faster scale physics is either subcycled or solved implicitly. Inherent limitations arise with semi-implicit integration as newly resolvable physics on finer grids is included.

Fully implicit (FI) numerical frameworks provide several potential benefits to large scale model development [3]. FI methods allow longer time steps or, conversely, finer resolution at a particular time step. The time step is chosen to resolve the physical processes of interest, which for climate models ranges from over a thousand years in the deep ocean to parameterized physics and chemistry occurring on the order of seconds. Enhanced accuracy is possible because all the terms in the model equations are solved coherently, and the time discretization determines the accuracy as the model is refined and time step increased. Also, the FI framework creates a pathway to determine model sensitivities through the exploration of parameter space.

Using a solver package that is under active development and maintenance allows access to a suite of mature solvers and has maximized portability and performance for large-scale multiphysics applications [4,5]. The recent development of ForTrilinos, an interface between the C++ solver code and Fortran within the Trilinos solver package, allows for a systematic implementation of new solver and analysis capability with the Fortran-based global climate modeling community. Climate models are large, complex, multiscale, multi-institutional efforts that maintain scores of developers working in tandem, so implementing a tested and benchmarked solver package with seamless interactions between the climate code is a clear benefit.

Until recently, accessing C++ from Fortran required considerable compiler- and platform-specific knowledge, including the chosen compilers' name mangling conventions and the hardware representation of each language's data types. Achieving portability further required "flattening" interfaces by exporting C++ member functions as external procedures and flattening data structures into one-dimensional lists, the elements of which had to be chosen from a small set of types sharing common bit representations in Fortran and C++ [6]. ForTrilinos capitalizes on recently expanded compiler support for Fortran 2003 [7], which simplifies the above process and increases its portability. The commonly supported features include object-orientation, the ability to bind C structs and function prototypes to their Fortran counterparts, and the ability to declare and manipulate C types natively in Fortran.

In this brief paper, we will demonstrate the virtually seamless presence of ForTrilinos with comparable results and scaling in atmospheric (HOMME) and ocean (POP) component climate models (sections 3 and 4) of the Community Climate System Model (CCSM) [8].

2 Interchangeable Solution Framework for a Global Climate Model

2.1 Nonlinearly Consistent Solution Algorithm

The Jacobian-Free Newton-Krylov (JFNK) fully implicit solver technique is used to integrate the model equations and illustrate the utility of the ForTrilinos framework. Future additions to the JFNK algorithm, including a suite of high performance preconditioners and analysis tools mentioned in section 2.2, as well as an extension to more realistic climate test cases, are ongoing within ForTrilinos. An extensive overview of the JFNK method is provided in Knoll and Keyes (2004) [9] so only a brief explanation is provided here.

The nonlinear partial differential equation (PDE) set can be written in residual form as a nonlinear operator, $\mathbf{F}$, on the vector of dependent variables, $\mathbf{x}$, of the problem to be solved. $\mathbf{F}(\mathbf{x})$ is evaluated at a time level and then with updates for $\mathbf{x}$ at some future time, given $\delta\mathbf{x}$,

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \delta\mathbf{x}^k, \tag{1}$$

until the residual decrease has reached a specified tolerance. The nonlinear iteration index k of the update is generated by solving a linearized version of the problem, a first-order Taylor series expansion of $\mathbf{F}$ about $\mathbf{x}^k$. The resulting Jacobian vector product is approximated with a finite difference approximation, rendering the operation "Jacobian-Free". The linear solution is found with the Generalized Minimum Residual Method (GMRES). The $\delta\mathbf{x}$ that satisfies the linear tolerance criterion, which is set constant to $\eta_l = 1 \times 10^{-6}$ for this study, is then sent to equation (1). The parameters used in this application of the JFNK solver in ForTrilinos are set to maximize efficiency while maintaining accuracy for the linear test case. The ability of JFNK to provide a scalable, efficient solution within multiscale codes depends on a quality, scalable preconditioner, a point illustrated in sections 3 and 4 below. To be consistent with the existing explicit and semi-implicit formulations in the component models, the explicit and fully implicit JFNK method are discretized in time with a second order method using leap-frog with an Asselin filter and Crank-Nicolson, respectively.

2.2 The Trilinos Framework of Scalable Solvers

The Trilinos framework of scalable solution algorithms [10] has a large set of mature solvers under active development, several of which have the potential to impact climate modeling. In terms of current capabilities, most Trilinos linear algebra algorithms are based on the "Epetra" data structure for vector and sparse matrix operations on distributed-memory parallel architectures. Several iterative linear solver algorithms, such as common variants of GMRES and CG solvers, can be invoked on Epetra data structures.

Above the linear algebra layer is the embedded analysis tool layer that includes (1) LOCA, a continuation package for parameter studies and bifurcation

analysis, (2) a time integration package, and (3) an iterative eigensolver for stability analysis. The JFNK algorithm described in section 2.1 has been implemented through the use of the nonlinear solve package NOX within LOCA. Exchanging layers to access these capabilities requires only small extensions to the interface for the nonlinear solver, and all of them make use of the Trilinos linear solvers underneath as needed.

The implementation of NOX within a climate model component provides not only an opportunity for improved accuracy and efficiency, but the enabling of a framework for the use of sophisticated analysis tools, including sensitivity and stability analysis, continuation and bifurcation studies, and the use of optimization algorithms to calibrate model parameters to data. Further, long term modeling development could benefit from several ongoing algorithm development efforts in Trilinos. Development of fast linear algebra kernels for multi-core and other modern computer architectures and work on new aggregation techniques for multi-level preconditioners that improve performance for non-symmetric systems that arise in highly-convective flows is ongoing. Also under development is a capability for solving for periodic orbits, which includes automatic discretization and parallelism over the time domain, which can be used, for example, to resolve steady annual cycles of ocean models.

2.3 Interfacing a Climate Component Code to Trilinos

Trilinos has been successfully connected with two component climate models. This process confronts two main hurdles: (1) Trilinos is written in C++, while climate codes are written in Fortran and (2) the solvers need to control the application. The first issue further breaks down into two subtopics: the language differences and the related programming paradigm differences. The component models employ procedural programming whereas Trilinos uses object-oriented programming in C++. For present purposes, the intersection between the climate and solver codes (one procedure call on either side) proved insufficient to justify simultaneously confronting the language and paradigm disparities. Hence, we focus on language interoperability and defer the discussion of object-orientation in Fortran 2003.

A general-purpose driver, `doloca`, facilitates invocation of the NOX nonlinear solver within the LOCA package described in section 2.2 in a procedural fashion. Portability derives from embedding a Fortran 2003 interface block of the form:

```
interface
  subroutine doloca(vector_size,vector,comm, &
                    vector_container,residual_calculator) &
  bind(C,name='doloca')
    use iso_c_binding, only : c_int,c_double,c_ptr,c_funptr
    integer(c_int)               :: vector_size,comm
    real(c_double), dimension(*) :: vector
    type(c_ptr)                  :: vector_container
    type(c_funptr), value        :: residual_calculator
```

```
    end subroutine
  end interface
```

This interface uses type parameters from the Fortran 2003 intrinsic module `iso_c_binding`. These parameters facilitate the use of similarly named C primitive types, including C pointers (`c_ptr`) and function pointers (`c_funptr`). Furthermore, the `value` attribute enables passing the corresponding argument by value instead of the Fortran default of passing by reference. Finally, the `bind(C,name='doloca')` construct binds the Fortran interface body to a corresponding C prototype and preserves case sensitivity on the C side via the `name` argument. As of December 2008, the compilers supporting these features include gfortran, g95 and those from Intel, IBM, Cray, Portland Group, and Pathscale.

The C++ code exports `doloca` via an `extern "C"` construct to suppress any C++-specific name mangling. (The aforementioned `bind` construct automatically handles any remaining name mangling.) The calling Fortran code encapsulates its solution vector and associated arguments needed to pass through the residual calculator subroutine in a Fortran derived type object. It then passes the size of the vector and the raw vector as the first and second argument to `doloca` respectively; the MPI communicator as the third argument; a C pointer to the derived type object as the fourth argument; and a C function pointer to a residual calculation procedure as the fifth argument. Inside `doloca`, NOX perturbs the raw vector and calls the residual calculator, passing it the vector and the derived type object, which is otherwise treated as a black box on the C++ side. Referencing the residual calculator via a C function pointer decouples the climate and solver source codes by obviating the need to hardwire the residual procedure name in the C++ source. Neither side dictates procedure names to the other, thus easing any later transition to a different solver or application.

The algorithmic issues of interfacing the C++ Trilinos framework and the Fortran coding are fairly simple. The vector object of the Epetra distributed parallel data structure can be constructed from an MPI Communicator, an integer length N_p of the vector on the current processor, and a double precision vector. These are all available in the Fortran code. Secondly, iterative implicit solver algorithms need to be able to call the code which provides the residual evaluation, $F(\mathbf{x})$, as a stateless subroutine (function call), given a solution vector. Unlike the implicit version, the explicit code has the evaluation of the PDE, time integration, and the update of the solution vector to the next time step all mixed together in the same code.

Presently, the climate codes are rewritten to formulate the residual of the function evaluation directly in terms of the solution vector and the time derivative of the solution vector. In POP, which is semi-implicit, this required creating a right hand side of the governing equations (refer to section 4). Both codes had their build system adapted to include the Trilinos libraries and compilation of multiple code languages.

3 JFNK Integration in HOMME Shallow Water Test Case

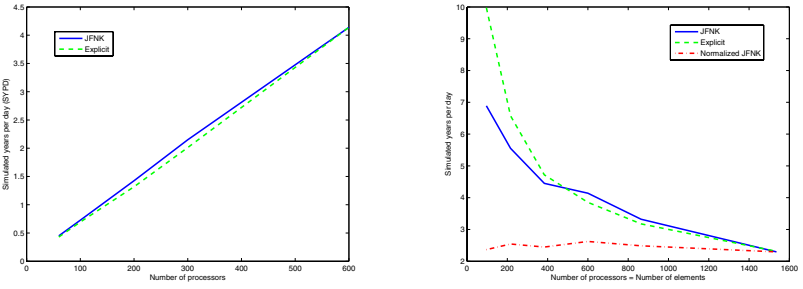
The HOMME atmospheric component model option of the CCSM uses a spectral element spatial discretization scheme, with a full description given elsewhere [11,12]. The cubed sphere uses a tiled, inscribed cube mapped to the sphere, which avoids the very disparate horizontal grid sizes near the poles that exist within a traditional latitude/longitude discretization. Ideally, the spectral element discretization on the cubed sphere retains the scalability and geometric flexibility of finite elements combined with superior accuracy and exponential convergence within each element, whereby the $O(N^3)$ cost associated with the Legendre transforms is mitigated with the use of a lower order discretization relative to a full sphere. This balance of accuracy and expense provides a scalable dynamic core option up to $O(100K)$ processors [2]. The discretization notation used here follows earlier convention, MxNxN, where M is the total number of elements on the sphere and N is the polynomial degree.

A suite of tests exist to evaluate the quality of numerical methods within a global scale atmospheric climate model [13], and the most basic test case is presented here to illustrate the utility of the ForTrilinos solver framework. The following linear example, named TC1, demonstrates the ability of HOMME to solve the shallow water equations with a fixed velocity profile. The test case specifies an initial cosine bell anomaly and the error is assessed after one rotation around the globe. In advective and residual form, the height equation is

$$0 = \frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{v}); \qquad \mathbf{v} = \{u, v\}^T, \tag{2}$$

where h is the depth of the fluid above the topography features in the model and the two-dimensional velocity field, $\mathbf{v}$, is specified in Williamson et al. (1992) in equations (75)-(76). Although only h is solved here, the full nonlinear solution framework outlined in section 2 is fully implemented. The specific parameters are matched to earlier TC1 runs with HOMME using the fully explicit [11] and semi-implicit methods [12], whereby the anomaly is advected over the corners of the cubed sphere edges using a fixed zonal rotation at an angle $\pi/4$ from the Earth's equatorial axis and the spatial grid is set to 96x16x16, which corresponds to 24576 grid points and an average resolution of about 167 km (minimum 38 km and corresponding CFL limit of 36 s).

The net time to solution for a 12 day simulation (throughout, day 1 is not included) of the explicit and JFNK method with a 30 s time step size on the Jaguar Cray XT4 on 96 processors is 4 min 53 s and 102 min 49 s with a final L_2 norm of error of 3 and 1.7×10^{-3} respectively. Because the JFNK method requires multiple function evaluations of the residual at each time step, it is more expensive than an explicit integration procedure using the same time step size. However, the time step size using JFNK is not limited by the time scale



(a) Strong Scaling for 600x16x16 (b) Scaling for Mx16x16, 1 element/proc

Fig. 1. Strong (a) and weak (b) scaling of TC1 benchmark problem for the explicit (green dashed line) and implicit (blue solid line) JFNK solution methods. SYPD is the simulated years per day. For (b), the explicit time step for 96x16x16 matches [12] and is reduced proportionally with grid refinement to maintain the same level below the stability limit. The red dashed line represents the simulation time for the normalized JFNK (see text), respectively. M is the number of elements on the sphere.

of the fastest waves in the system. When run with a 12 minute (720 s) time step size, the error using JFNK is larger than explicit (8.5×10^{-3}), however the JFNK simulation time is significantly reduced to 7 min 1 s. Note that the JFNK method is currently not preconditioned, which would reduce the simulation time and number of function evaluations further with no loss in accuracy.

The long term goal is to design a scalable method for finer resolution climate modeling studies. Figure 1a shows the strong scaling of the ForTrilinos implementation of JFNK in HOMME in units of simulated years per day (SYPD), and it is clear the method scales well up to the limits of the decomposition, one element per processor for a finer 600x16x16 grid (avg. spacing of 66.8 km). At this resolution and a 720 s time step, the JFNK method takes about the same time to complete the simulation as the explicit method. Figure 1b displays the SYPD for range of grids decomposed at 1 element per processor. This is a weak scaling without an increase in the domain, so a necessary refinement of the grid requires a smaller time step size for the explicit method and a corresponding increase in simulation time, or decreased SYPD (green dashed line). Using the JFNK method, the simulations can be completed using the same time step size with grid refinement, however they experience the same increase in simulation time (blue solid line). This is associated with the increased number of linear iterations by the Krylov-based linear solver within the outer nonlinear iteration [14]. To illustrate this effect, the SYPD for each run is normalized to the finest grid using a ratio of average number of iterations at the current and finest grid. The weak scaling of the normalized JFNK simulations (red dot-dashed line) is mostly flat with grid refinement. The action of a preconditioner is to flatten the blue JFNK line at some level above the dashed red line.

4 JFNK Integration in POP Channel Test Case

The JFNK implementation within HOMME has also been implemented in the Parallel Ocean Program (POP), a state-of-the-art Ocean General Circulation Model that is routinely run on a global domain at 0.1° spatial resolution [15], and is the ocean component of the CCSM. With implicit time stepping being implemented simultaneously in both the atmosphere and ocean components of CCSM, we are working towards a coupled system where potential efficiency gains can be obtained in its two most expensive components. Currently in POP, the (fast) barotropic mode is split off, and solved implicitly. The remaining baroclinic system is solved using an explicit leap-frog scheme. Still, increased resolution puts a severe limit on the time step that can be taken, even when no additional physical scales are included in the model.

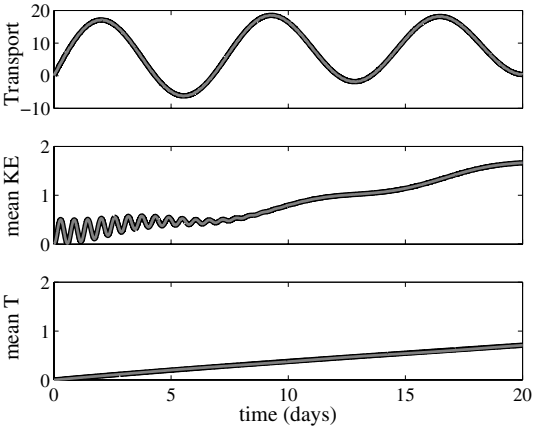


Fig. 2. Channel transport (in $10^6\,\mathrm{m}^3\,\mathrm{s}^{-1}$), mean kinetic energy ($\mathrm{kg}\,\mathrm{m}^{-1}\,\mathrm{s}^{-2}$), and mean temperature (in $10^{-3}K$), for a simulation of the POP model in a reentrant channel configuration. The JFNK scheme (gray) matches the conventional POP time stepping scheme (black).

In implementing implicit methods in POP, like HOMME, our main goals were to invade the core code as little as possible, make it user-friendly, and make it work for most, if not all, of the available physics options. Our implementation consisted of 2 major developments. In the first phase, infrastructure was added to allow for arbitrary time-stepping schemes without mode splitting, like 4th-order Runge Kutta. This required adding capability to calculate tendencies of the prognostic variables given a state vector, and to construct appropriate right-hand side, or $F(\mathbf{x})$, similar to [16]. Some extra routines and runtime switches were added, but required only minor modifications to the core POP code. The second phase of the development was specific to implicit stepping. It consisted of coupling POP to the ForTrilinos framework as outlined in section 2.3 to access the JFNK solution algorithms in Trilinos.

To test the implementation of the implicit solver in POP, we used a simple test case of the reentrant channel with an undersea bump [17]. The model is forced by a wind stress, as well as a restoring of Sea-Surface Temperature (SST) to a prescribed profile that changes linearly over the width of the channel. Figure 2 shows that the JFNK method with Crank-Nicolson implicit time stepping produces the same solution as with the leap-frog time stepping. Thus using the Trilinos solver package to integrate implicitly in POP can be achieved with little change to the underlying climate component models. The JFNK method, like HOMME, will not outperform the conventional method explicit method in terms of runtime without preconditioning.

5 Summary

The implementation of the ForTrilinos interface to the Trilinos solver package allows two Fortran-based climate components of the CCSM to take advantage of a suite of high fidelity solution methods. ForTrilinos successfully reproduces solutions to the corresponding test cases without significant alteration of the code structure or degradation of the scaling behavior. Note also that the JFNK method is called within the Trilinos LOCA package, so it is now possible to perform analyses such as parameter continuation on the model with no additional solver coding or computational expense. Also, because the JFNK method allows larger time step sizes, refined versions of the grids used for a simple benchmark study completes the simulation in the same time as the explicit method. Weak scaling of HOMME using the JFNK method performs like explicit, with increased simulation time upon grid refinement, but for different reasons. Unlike the explicit method, the JFNK method has a way around this issue, which is to reduce the number of linear iterations using a scalable preconditioner. With a good preconditioner, previous work has shown minimal growth of linear iterations with problem size [18]. This is the focus of future work using the JFNK method in component climate models within the CCSM.

Acknowledgments

The authors would like to thank Mike Heroux and Roger Pawlowski for their enabling contributions. This project was funded by the Office of Science within the U.S. Department of Energy (DOE-OS) through a combination of the Oak Ridge National Laboratory Laboratory Directed Research and Development program (Evans and White), the Climate Change Prediction Program (Weijer), and the TOPS II project within SciDAC (Rouson and Salinger), a Division of the Office of Advanced Scientific Computing Research in DOE-OS. We also acknowledge the Core Development Group of the HOMME project, which is part of the National Science Foundation funded National Center for Atmospheric Research and the POP developers. This research used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by DOE-OS under Contract DE-AC05-00OR22725.

References

1. Simon, H., Bader, D.A. (eds.): Software Deisgn for Petascale Climate Science. In: Petascale Computing: Algorithms and Applications, ch. 7. Chapman and Hall/CRC, Boca Raton (2008)
2. Taylor, M.A., Edwards, J., St-Cyr, A.: Petascale atmospheric models for the community climate system model: new developments and evaluation of scalable dynamic cores. J. Phys. Conf. Series 125, 012023 (2008)
3. Keyes, D., Reynolds, D., Woodward, C.: Implicit solvers for large-scale nonlinear problems. J. Phys. Conf. Series 46, 433–442 (2006)
4. Lasater, M., Kelley, C., Salinger, A., Wollard, D., Zhao, P.: Parallel parameter study of the Wigner-Poisson equations for RTD's. Comp. Math. App. 51, 1677–1688 (2006)
5. Chacon, L.: Parallel implicit solvers for 3D magnetohydrodynamics. J. Phys. Conf. Series 125, 012041 (2008)
6. Gray, M.G., Roberts, R.M., Evans, T.M.: Shadow-object interface between Fortran 95 and C++. Comp. Sci. Eng. 1(2), 63–70 (1999)
7. Chivers, I.D., Sleightholme, J.: Compiler support for the Fortran 2003 standard. ACM Fortran Forum 27(2) (2008)
8. Collins, W.D., et al.: The Community Climate System Model Version 3 (CCSM3). J. Climate 19, 2122–2143 (2006)
9. Knoll, D., Keyes, D.: Jacobian-free Newton-Krylov methods: A survey of approaches and applications. J. Comput. Phys. 193, 357–397 (2004)
10. Heroux, M.A., Bartlett, R.A., Howe, V.E., Hoekstra, R.J., Hu, J.J., Kolda, T.G., Lehoucq, R.B., Long, K.R., Pawlowski, R.P., Phipps, E.T., Salinger, A.G., Thornquist, H.K., Tuminaro, R.S., Willenbring, J.M., Williams, A.: An overview of the Trilinos project. ACM Trans. Math. Soft. 31(3), 397–423 (2005)
11. Taylor, M.A., Tribbia, J., Iskandarani, M.: The spectral element method for the shallow water equations on the sphere. J. Comput. Phys. 130, 92–108 (1997)
12. Thomas, S.J., Loft, R.D.: Semi-implicit spectral element model. SIAM J. Sci. Comput. 17, 339–350 (2002)
13. Williamson, D.L., Drake, J.B., Hack, J.J., Jakob, R.J., Swarztrauber, P.: A standard test set for numerical approximations to the shallow water equations in spherical geometry. J. Comput. Phys. 102, 211–224 (1992)
14. Iskandarani, M., Haidvogel, D.B., Boyd, J.P.: A staggered spectral element model with application to the oceanic shallow water equations. Internat. J. Numer. Methods Fluids, 393–414 (1995)
15. Maltrud, M.E., McClean, J.L.: An eddy-resolving global $1/10^{o}$ ocean simulation. Ocean Modeling 8, 31–54 (2005)
16. Nadiga, B.T., Taylor, M.A., Lorenz, J.: Ocean modelling for climate studies: Eliminating short time scales in long-term, high-resolution studies of ocean circulation. Math. Comp. Mod. 44, 870–886 (2006)
17. Petersen, M.R., Hecht, M.W., Wingate, B.A.: Efficient form of the lans-α turbulence model in a primitive-equation ocean model. J. Comput. Phys. 227, 5717–5735 (2008)
18. Evans, K., Knoll, D.A., Pernice, M.A.: Enhanced algorithm efficiency using a multigrid preconditioner and SIMPLE based smoother. J. Comput. Phys. 223, 121–126 (2007)

GeoComputation 2009

Yong Xue[1,2], Forrest M. Hoffman[3], and Dingsheng Liu[4]

[1] State Key Laboratory of Remote Sensing Science, Jointly Sponsored by the Institute of Remote Sensing Applications of Chinese Academy of Sciences and Beijing Normal University, Institute of Remote Sensing Applications, Chinese Academy of Sciences, P. O. Box 9718, Beijing 100101, China
[2] Department of Computing, London Metropolitan University, 166-220 Holloway Road, London N7 8DB, UK
[3] Oak Ridge National Laboratory, Computational Earth Sciences Group, Building 5600, Room C221, MS 6016, P.O. Box 2008, Oak Ridge TN 37831-6016, USA
[4] Center for Earth Observation and Digital Earth, Chinese Academy of Sciences, No.45, Bei San Huan Xi Road, Beijing, China
y.xue@londonmet.ac.uk, forrest@climatemodeling.org, dsliu@ceode.ac.cn

1 Preface

The tremendous computing requirements of today's algorithms and the high costs of high-performance supercomputers drive us to share computing resources. The emerging computational Grid technologies are expected to make feasible the creation of a computational environment handling many PetaBytes of distributed data, tens of thousands of heterogeneous computing resources, and thousands of simultaneous users from multiple research institutions (Giovanni *et al.* 2003).

The Workshop on GeoComputation continues with the ICCS conferences held in Amsterdam (2002), St. Petersburg (2003), Krakow (2004), Atlanta (2005), Reading (2006), Beijing (2007) and Krakow (2008). GeoComputation is about using various different types of geographical and environmental data and developing relevant tools within the overall context of a computational scientific approach. It is concerned with new computational techniques, algorithms, and paradigms that are dependent upon and can take advantage of Grid Computing. It includes spatial data analysis, dynamic modeling, simulation, space-time dynamics and visualization and virtual reality. This conference will offer presentations from a variety of sources, both local, national and international and will enable you to network with others working in similar fields.

Grid computing technology is a new method for processing remotely sensed data. Jianwen Ai et al. in their paper "Grid Workflow Modeling for Remote Sensing Retrieval Service with Tight Coupling" discusses some application cases based on Grid computing for Geo-sciences and the application limit of Grid in remote sensing, and provides a method for Grid Workflow modeling for remote sensing. Tight-coupling remote sensing algorithms cannot be scheduled by a Grid platform directly. Therefore, we need an interactive graphical tool to present the executing relationships of algorithms and to generate automatically the corresponding submitted description files for a Grid platform.

Image resampling, which is frequently used in remote sensing processing procedures, is a time-consuming task. Parallel computing is an effective way to speed up

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 345–348, 2009.

this processing; however, recent parallel image resampling algorithms with massive time-consuming global processes like I/O, always lead to low efficiency and non-linear speedup ratios, especially when the number of computing nodes increases beyond a certain extent. And what's more, the various geo-referencing related to different processing applications cause a real problem for code reuse. To solve these problems, PIRA-PIO (Parallel Image Resampling Algorithm with Parallel I/O) algorithm, an asynchronous parallelized image resampling algorithm with parallel I/O, is proposed in the paper by Ma *et al.* "An Asynchronous Parallelized and Scalable Image Resampling Algorithm with Parallel I/O". Parallel I/O on parallel file systems and asynchronous parallelization using I/O hidden policy to sufficiently overlap the computing time with I/O time is used in PIRA-PIO for performance enhancement. In addition, the design of reusable code like design pattern will be used for improving flexibility in different remote sensing image processing applications. Through experimental and comparative analysis, its outstanding parallel efficiency and perfect linear speedup is shown in this paper.

Jingshan Li *et al*. in their paper "Design and Implementation of a Scalable General High Performance Remote Sensing Satellite Ground Processing System on Performance and Function" discuss design and implementation of a scalable high performance remote sensing satellite ground processing system using a variety of advanced hardware and software application technology for performance and function. These advanced technologies include the network, parallel file system, parallel programming, job scheduling, workflow management, design patterns, etc., which make the performance and function of remote sensing satellite ground processing systems scalable enough to fully meet the high performance processing requirements of multi-satellite, multi-tasking, massive remote sensing satellite data. The "Beijing-1" satellite remote sensing ground processing system is introduced as an instance.

Remote sensing data plays a key role in understanding complex geographic phenomena. Clustering is a useful tool in discovering interesting patterns and structures within multivariate geospatial data. One of the key issues in clustering is the specification of an appropriate number of clusters, which is not obvious in many practical situations. In this paper "Incremental Clustering Algorithm for Earth Science Data Mining" Ranga Raju Vatsavai provided an extension of a G-means algorithm which automatically learns the number of clusters present in the data and avoids over estimation of the number of clusters. Experimental evaluation on simulated and remotely sensed image data shows the effectiveness of their algorithm.

The booming of Earth observation provides decision-makers with more available geospatial data as well as more puzzles about how to understand, evaluate, search, process, and utilize those overwhelming resources. The paper from Wang *et al.* "Overcoming Geoinformatic Knowledge Fence: An exploratory of intelligent geospatial data preparation within spatial analysis" introduce a concept termed geoinformatic knowledge fence (GeoKF) to discuss the knowledge-aspect of such puzzles and an approach to overcoming them. Based on analysis of the gap between common geography sense and geoinformatic professional knowledge, the approach comprises analysis of space modeling and spatial reasoning to match decision models to the online geospatial data sources they need. Such approaches enable automatic and intelligent searching of suitable geospatial data resources and calculating their suitability to a given spatial decision and analysis. An experiment with geo-services, geo-ontology and rule-based reasoning

(Jess) is developed to illustrate the feasibility of the approach in scenarios of data preparation within decisions of bird flu control.

Service Oriented Architecture is not widely used in GIS. Although there are a few organizations that have launched Service Oriented GIS applications, it is not possible for all interested parties to utilize those services. Because those are proprietary organizations, users have to pay large amounts of money for those services. Sometimes they do not always provide the desired services to the users either. Therefore users have to move to another application. It is really a waste of money and time. The paper "Service Oriented Customizable Framework to Manipulate GIS Data" from Ranasinghe and Karunaratne aims to provide a solution by designing a service oriented customizable framework to manipulate GIS data.

Spatial relations play an important role in computer vision, scene analysis, geographic information systems (GIS) and content-based image retrieval. Fuzzy Allen relations are used to define the fuzzy topological relations between different objects and to detect object positions in images. In the paper "Spatial Relations Analysis by Using Fuzzy Operators" from Salamat and Zahzah, fuzzy aggregation operators are used for information integration along with polygonal approximation of objects. This new approach offers low temporal and computational complexity for the extraction of topological and directional relations.

A wide variety of data mining techniques are being applied to the growing body of Earth Science data. From small scale measurement data to global climate simulation output, very large or long time series databases of environmental and climate data are proving difficult to analyze and interpret. Data mining techniques--like cluster analysis, principle components analysis (PCA), classification and regression tree (CART) analysis, and neural networks--are being applied to problems of feature extraction, model-data comparison, and validation/verification. However, the size and complexity of Earth Science data are stretching the limits of commercial statistical packages and many freely available analysis tools, which were not designed to scale up to terabyte- and petabyte-sized datasets. Scalable statistical tools are needed to run on very large parallel supercomputers in order to analyze data of this size. The following papers address these issues by demonstrating how data mining techniques can be applied in the Earth Sciences and by describing innovative computer science techniques and methods that can support analysis and discovery in Earth Sciences.

Increasingly large datasets acquired by NASA for global climate studies demand larger computation memory and higher CPU speed to extract useful and revealing information. While boosting the CPU frequency is getting harder, clustering multiple lower performance computers thus becomes increasingly popular. This prompts a trend of parallelizing the existing algorithms and methods by mathematicians and computer scientists. In the paper "A Parallel Nonnegative Tensor Factorization Algorithm for Mining Global Climate Data", Zhang *et al.* take on the task of parallelizing the Nonnegative Tensor Factorization (NTF) method, with the purpose of distributing large datasets across cluster nodes, thus reducing the demand on a single node, blocking and localizing the computation at the maximal degree, and finally minimizing the memory use for storing matrices or tensors by exploiting their structural relationships. Numerical experiments were performed on a NASA global sea surface temperature dataset and resulting factors are analyzed and discussed.

The ultimate goal of data visualization is to clearly portray features relevant to the problem being studied. This goal can be realized only if users can effectively

communicate to the visualization software important features of interest. To this end, Johnson *et al.* describe in the paper "Querying for Feature Extraction and Visualization in Climate Modeling" two query languages used by scientists to locate and visually emphasize relevant data in both space and time. These languages offer descriptive feedback and interactive refinement of query parameters, which are essential in any framework supporting queries of arbitrary complexity. They apply these languages to extract features of interest from climate model results and describe how they support rapid feature extraction from large datasets.

The recurrence of periodic environmental states is important to many systems of study, and particularly to the life cycles of plants and animals. Periodicity in parameters that are important to life, such as precipitation, are important to understanding environmental impacts, and changes to their intensity and duration can have far reaching impacts. To keep pace with the rapid expansion of earth science datasets, efficient data mining techniques are required. Discrete Fourier transform (DFT) and wavelet analysis are useful data mining tools for rapidly searching for changes in the intensity of seasonal, annual, or interannual events by projecting the magnitude and shift of periodicities onto power spectrum plots. Brooks explores the strengths and limitations of DFT and wavelet spectral analysis using output from the Parallel Climate Model (PCM). Spectral analysis is used to diagnose model behavior, and locate land surface cells that show shifting cycle intensity, which could be used as an indicator of climate change. Example routines in Octave/Matlab and IDL are provided.

Danek's paper "Seismic wave field modeling with graphics processing units" describes the GPGPU - general-purpose computing on graphics processing units, which is a very effective and inexpensive way of dealing with time consuming computations. In some cases even a low end GPU can be a dozens of times faster than a set of modern CPUs. Utilization of GPGPU technology can make a typical desktop computer powerful enough to perform necessary computations in a fast, effective and inexpensive way. Seismic wave field modeling is one of the problems of this kind. Sometimes one modeled common shot-point gather or one wave field snapshot can reveal the nature of an analyzed wave phenomenon. On the other hand these kinds of models are often a part of complex and extremely time consuming methods with almost unlimited needs for computational resources. This is always a problem for academic centers, especially now when times of generous support from oil and gas companies have ended.

Acknowledgment

We would like to thank our reviewers, Stefano Furin (Università di Ferrara, Dipartimento di Scienze della Terra), William W. Hargrove (U.S. Department of Agriculture – Forest Service), Jian Huang (University of Tennessee), Phillip Kegelmeyer (Sandia National Laboratories), Vipin Kumar (University of Minnesota), G.Q. Li (CAS, China), Kamesh Madduri (Lawrence Berkeley National Laboratory), Richard T. Mills (Oak Ridge National Laboratory), George Ostouchov (Oak Ridge National Laboratory), Christopher T. Symons (Oak Ridge National Laboratory), Raju R. Vatsavai (Oak Ridge National Laboratory), and Xingquan "Hill" Zhu (Florida Atlantic University), for sharing their kind support and expertise to improve the content of this workshop.

Grid Workflow Modeling for Remote Sensing Retrieval Service with Tight Coupling

Jianwen Ai[1,3,4], Yong Xue[1,2], Jie Guang[1,4], Yingjie Li[1,4], Ying Wang[1,4], and Linyan Bai[1,4]

[1] State Key Laboratory of Remote Sensing Science, Jointly Sponsored by the Institute of Remote Sensing Applications of Chinese Academy of Sciences and Beijing Normal University, Institute of Remote Sensing Applications, Chinese Academy of Sciences, P.O. Box 9718, Beijing 100101, China
[2] Department of Computing, London Metropolitan University, 166-220 Holloway Road, London N7 8DB, UK
[3] College of Resources and Environmental Sciences, Northeast Agricultural University, Harbin, 150030, China
[4] Graduate University of Chinese Academy of Sciences, Beijing 100049, China
neau_ajw@hotmail.com, y.xue@londonmet.ac.uk

Abstract. Grid computing technology is a new way for remotely sensed data processing. Tight-coupling remote sensing algorithms can't be scheduled by grid platform directly. Therefore, we need a interactive graphical tool to present the executing relationships of algorithms and to generate automatically the corresponding submitted description files for grid platform. In this paper we mainly discusses some application cases based on Grid computing for Geosciences and the application limit of Grid in remote sensing, and gives the method of Grid Workflow modeling for remote sensing. Then based on the modeling, we design a concrete example.

1 Introduction

Remote sensing data is characterized by largeness and instantaneousness. The analysis and sharing of these huge amounts of data is a big challenge for the remote sensing community (Hu *et al.* 2005). The tremendous computing requirement of the algorithms and the high costs of high-performance supercomputers drive us to hunt for share of computing resources. The emerging computational grid technologies are expected to make feasible the creation of a computational environment handling many PetaBytes of distributed data, tens of thousands of heterogeneous computing resources, and thousands of simultaneous users from multiple research institutions (Giovanni et al. 2003).

Fortunately, within the spatial information field, there are successful application cases based on Grid computing. Work Package (WP) 9 of the DataGrid aims to demonstrate the use of Grid technology for remote sensing applications and earth observation (Giovanni *et al. 2003*). The Information Power Grid (IPG) (http:// www. ipg.nasa.gov) is NASA's high-performance computational Grid. Computational Grids are persistent networked environments that integrate geographically distributed

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 349–356, 2009.

supercomputers, large databases, and high-end instruments. These resources are managed by diverse organizations in widespread locations and shared by researchers from many different institutions (http://www.ipg.nasa.gov). The IPG is a collaborative effort between NASA Ames, NASA Glenn, and NASA Langley Research Centers, and the NSF PACI programs at SDSC and NCSA (http://www.ipg.nasa.gov). GENIE (Grid ENabled Integrated Earth System Model) is a new Grid-enabled modeling framework that can compose an extensive range of Earth System Models (ESMs) for simulation over multi-millennial timescales, to study ice age cycles and long-term human-induced global change (Andrew *et al. 2005*). The scientific focus of GENIE is on long-term and paleo-climate change, especially through the last glacial maximum (~21kyr BP) to the present interglacial, and the future long-term response of the Earth system to human activities (http://www.genie.ac.uk/about/overview.htm. The goal of GENIE is to integrate models of the atmosphere, ocean, sea-ice, marine sediments, land surface, vegetation and soil, ice sheets and the energy, biogeochemical and hydrological cycling within and between components (http://www.genie.ac.uk/about/modelling.htm). The Earth System Grid II (ESG) (http://www.earthsystemgrid.org/about/ overviewPage.doc) is a new research project sponsored by the U.S. DOE Office of Science under the auspices of the Scientific Discovery through Advanced Computing program (SciDAC). The primary goal of ESG is to address the formidable challenges associated with enabling analysis of and knowledge development from global Earth System models (http://www.earthsystemgrid.org/about/overviewPage.doc). GEON (GEOsciences Network) is the cyberinfrastructure project that is bringing together information technology and geoscience researchers from multiple institutions in a large-scale collaboration (http://www.geongrid.org). The aim of GEON is to build data-sharing frameworks, identify best practices, and develop useful capabilities and tools to enable dramatic advances in how geoscience is done (Young *et al. 2005*). TeraGrid is a collaboration of partners providing a high-performance, nationally distributed capability infrastructure for computational science (Charles 2005). The Geographic Information Science Gateway (GISolve) is a TeraGrid Science Gateway project based at the National Center for Supercomputing Applications (NCSA) (http://kb.iu.edu/data/awod.html). Its focus is on geographic information science, an interdisciplinary field involving geography and other social sciences, computer science, geodesy, and information sciences for the study of generic issues in the development and use of computationally intensive geographic information systems (GIS) technologies (http://kb.iu.edu/data/awod.html).

2 The Application Limit of Grid in Remote Sensing

Remote sensing quantitative retrieval is a complex computing process due to the terabytes or petabytes of data processed and the tight-coupling remote sensing algorithms. The tight-coupling feature makes that remote sensing algorithm modules need to be processed by computer according to the logic order. The transfer among sensing algorithm modules not only includes processed data files, but also includes control files. Real largeness remote sensing data movement between remote sensing algorithm modules scheduled must be either via an interaction with a data movement service, or through specialized binary-level data channel running directly

between the tasks involved (Fox and Gannon 2006). The feature makes that the existing Grid platform cannot satisfy with our requirements. When we use Grid platform schedule sensing algorithm modules directly, we find that we cannot get the expectable result. Grid platform can only schedule irrelevant job. Therefore, we need to design a tool to control the order of remote sensing algorithms scheduled in Grid system. Besides, although much of Grid software technology addresses the issues of resource scheduling, quality of service, fault tolerance, decentralized control and security and so on, which enable the Grid to be perceived as a single virtual platform by the user, grid computing is not yet mature (Berman *et al. 2003*). There are many open issues to be addressed and missing functionality to be developed, and more will emerge as uses of computing Grids proliferate (Berman *et al. 2003*). Grid platform is not a special platform for remote sensing. It can also not make validity check of condition for remote sensing algorithm modules to be scheduled with short of special knowledge on remote sensing.

To solve the above problem, we must let the remotely sensed data processing module be scheduled in the Grid environment. We need design an interactive GUI interface to represent the processed steps of remote sensing algorithm modules and their relations including concurrence/synchronism and executed order. We need a tool which can use interactive graphical editors to present the executing relationships of algorithms on human-friendly diagrams and to generate automatically the corresponding submitted description files of grid platform. Fortunately, workflow technologies make it possible. Using workflow technology, we can construct a remote sensing information processing environment to integrate the distributed data and computational resources. It is not a new idea to apply workflow technologies to Grid platform. Actually, people try to integrate the workflow technologies under Grid platforms in many projects, such as Triana (Majithia et al. 2004), Unicore (Riedel et al. 2006), Kepler (Zhang 2006), ICENI (McGough et al. 2006), Taverna (Turi et al. 2007), GridFlow (Cao et al. 2002, 2003), Askalon (Fahringer et al. 2005), Karajan (http://www.cogkit.org), etc. These Grid workflows give a host of useful workflow composition tools with graph-based modeling or language-based modeling. About Language-based modeling, Yu and Buyya (2005) consider that language-based modeling may be convenient for skilled users, but they require users to enumerate a lot of language specific syntax; in addition, it is impossible for users to express a complex and large workflow by scripting workflow components manually; and workflow languages are more appropriate for sharing and manipulation, whereas the graphical representations are intuitive but they require to be converted into other forms for manipulation. So most Grid systems, workflow languages are designed to bridge the gap between the graphical clients and the Grid workflow execution engine (Guan et al. 2004). By analyzing, we find that these existing Grid workflow platforms can not satisfy with our requirement owing to the features of remote sensing, such tight-coupling remote sensing algorithm modules, largeness and instantaneousness remote sensing data, etc. Therefore, it is necessary to design a workflow composition tool using graph-based modeling for remote sensing services.

3 Grid Workflow Modeling for Remote Sensing Retrieval Service

Grid workflow modeling for remote sensing retrieval service includes that its task definition, structure definition and mapping relation of specific Grid resources for task execution. Task definition includes that all information to execute a task in grid environment, such as function descriptions of task, previous task, support environment requirements, the size of memory, minimum space of hard disk, etc. In general, a workflow structure can be represented as a Directed Acyclic Graph (DAG) or a non-DAG (Sakellariou and Zhao 2004). Non-DAG workflow includes the iteration structure, which isn't suitable for modeling for remote sensing algorithm modules. In DAG-based workflow, Yu and Buyya (2005) consider that workflow structure can be classified as sequence, parallelism, and choice; Sequence is defined as an ordered series of tasks, with one task starting after a previous task has completed; Parallelism represents tasks which are performed concurrently, rather than serially; and in choice control pattern, a task is selected to execute at runtime when its associated conditions are true. Mapping relation of specific Grid resources for task execution binds workflow tasks to specific grid resources. Aiming at largeness and instantaneousness of remote sensing data, we adopt tasks acting as data movement or computing code movement according to the schedule arithmetic of engine. Besides, grid workflow modeling for remote sensing retrieval service, unlike the object-orient design where interaction of the objects are driven by method calls, the states of our model can control their own actions and react to parameters which are the control-flow elements such as branching or an expression of value provided by their users. It also provides a mechanism for concurrency or sequential computation through tokens that are fired by the transition function. When there is a transition of the state, the model get the parameter of tokens from a FIFO (first in, first out) queue with capacity equal to one, executing the concurrency or sequential computation correspondingly. The Grid workflow modeling for remote sensing retrieval service can be represented as an 8-tuples GWP = (K, D, R, P, s, d, F, T), where:

- K is a finite set of states, where each black-box is regarded as a state. Any a element of K expresses a remote sensing algorithm module or a task of remote sensing data movement or remote sensing algorithm code movement.
- D is a set of data, which includes the initial data file of remote sensing information, the results of data disposed. D is the condition of remote sensing algorithm modules executed.
- R is a subset of binary relation K x K. It is a set of arcs, where each element represents the order relation among the executed remote sensing algorithm modules.
- P is a finite set of mapping parameters, parameters provided by users and control tokens. The mapping relation of specific Grid resources for remote sensing algorithm module execution, which binds workflow tasks to specific grid resources. The parameters provided by users include function descriptions of task, previous task, support environment requirements, the size of memory, minimum space of hard disk, etc. The control tokens expresses that remote sensing algorithm module gets the condition scheduled.
- $s \in K$ and $d \in D$ are the initial remote sensing state and initial remote sensing data file.

- F $\subseteq$ K is the set of final states, and
- T is a transition function from (K-F) x (P $\cup$ { Φ }) x D to K x D.
- We now formalize the operation of the model.

When the Workflow is started, triple (s, Φ ,d) (K $\subseteq$ F) x (P $\cup$ { Φ }) is its initial transition state of T; for all q$\in$ (K-F) and p$\in$ K, if (q, p)$\in$ R, the workflow scans its set of data and its set of parameters, getting relevant d1$\in$ D and p1$\in$ P; then the transition is fired, according to the semantic analysis of the parameters, and changing the state q to the state p and producing the result d2 from d1, until it finds a state pi$\in$ F; and it halts. The data file di is the result we need.

4 Implementation

We have implemented Grid workflow modeling for remote sensing retrieval service and Grid workflow composition tool using graph-based modeling for remote sensing services (see Figure 1). We can use XML to record the user's describe information of workflow (see Figure2). We have accomplished the scheduled of algorithms of remote sensing according workflow mechanism. The next work is to transform the XML format of workflows into the corresponding submitted description files of grid platform according to the criterions of Grid platform.

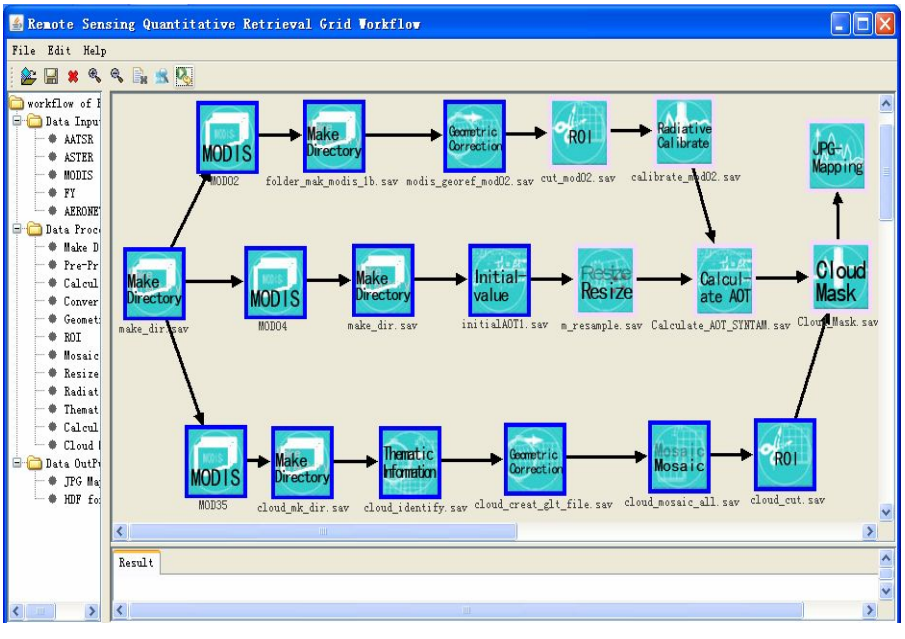


Fig. 1. A case of workflow scheduled

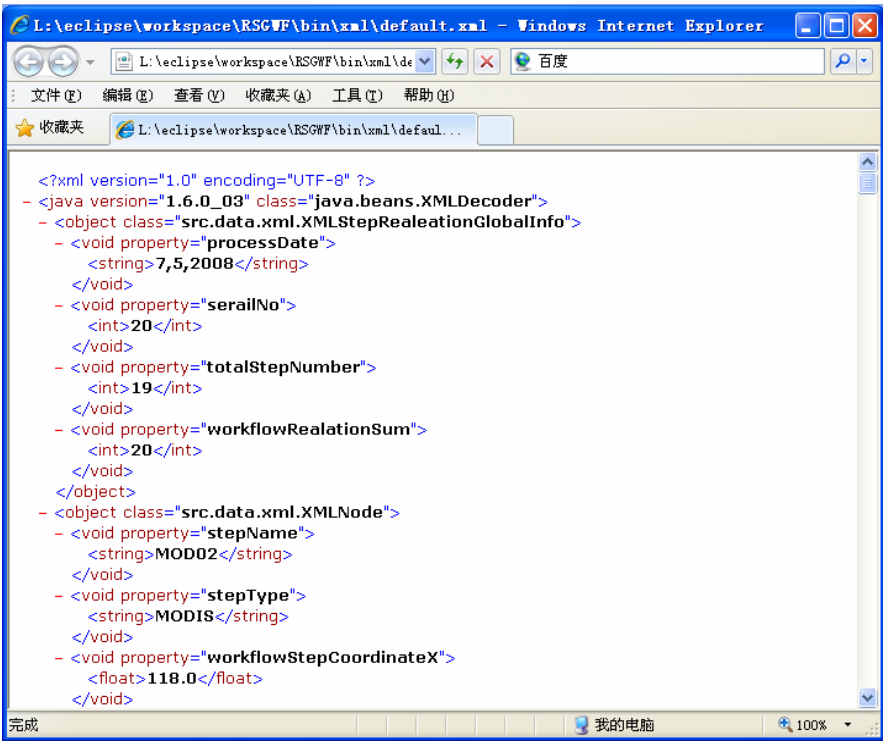


Fig. 2. A case of workflow xml format

5 Conclusion

Grid workflow composition tool using graph-based modeling can express the tight-coupling features of among remote sensing algorithms conveniently. Through it the existing remotely sensed data processing modules can be scheduled orderly by Grid platform. Through it, the existing remotely sensed data processing modules and algorithms resource can be shared. It seems like the common service in Grid environment shared. It much enhances the resource utility rate. In order to design a Grid workflow modeling for remote sensing retrieval service with tight coupling, the paper analyses the features of remote sensing data and algorithms and introduces the successful application cases based on Grid computing for Geo-sciences. By analyzing the application limit of Grid in remote sensing, we give Grid Workflow modeling for remote sensing retrieval Service. Combing the rule of modeling, we design Grid workflow composition tool using graph-based modeling for remote sensing services.

Acknowledgement

This work was supported in part by National Science Foundation of China (NSFC) under Grant No. 40671142, by the Ministry of Science and Technology (MOST),

China under Grant No. 2008AA12Z109 and Grant No. 2007CB714407, by Chinese Academy of Sciences (CAS) under Grant No. KZCX2-YW-313, by the NSFC under Grant No. 40471091.

References

[1] Xue, Y., Wang, J.Q., Wu, C.L., Hu, Y.C., Guo, J.P., Zheng, L., Wan, W., Cai, G.Y., Luo, Y., Zhong, S.B.: Information Registry of Remotely Sensed Meta-modeling Grid Environment. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3993, pp. 1–8. Springer, Heidelberg (2006)

[2] Fran, B., Hey Anthony, J.G., Fox Geoffrey, C.: Grid Computing Making the Global Infrastructure a Reality. John Wiley & Sons Ltd., UK (2003)

[3] Giovanni, N.A., Luigi, F.B., Linford, J.: Grid technology for the storage and processing of remote sensing data: description of an application. In: Proceedings of the society of photo-optical instrumentation engineers (SPIE), vol. 4881, pp. 677–685 (2003)

[4] Price, A., Lenton, T., Cox, S., Valdes, P., Shepherd, J., GENIE team: GENIE: Grid ENabled Integrated Earth System Model. ERCIM News 61, 15–16 (2005)

[5] Youn, C., Baru, C., Bhatia, K., Chandra, S., Lin, K., Memon, A., Memon, G., Seber, D.: GEONGrid Portal: Design and Implementations. In: GCE 2005 Workshop on Grid Computing based on SC 2005, November 2005, Seattle, WA (2005)

[6] Catlett, C.E.: TeraGrid: A Foundation for US Cyberinfrastructure. In: Jin, H., Reed, D., Jiang, W. (eds.) NPC 2005. LNCS, vol. 3779, p. 1. Springer, Heidelberg (2005)

[7] Yincui, H., Yong, X., Jiakui, T., Shaobo, Z., Guoyin, C.: Data-parallel Georeference of MODIS Level 1B Data Using Grid Computing. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3516, pp. 883–886. Springer, Heidelberg (2005)

[8] Majithia, S., Shields, M., Taylor, I., Wang, I.: Triana: a graphical Web service composition and execution toolkit. In: Proceedings of Web Services 2004, pp. 514–521 (2004)

[9] Riedel, M., Menday, R., Streit, A., Bala, P.: A DRMAA-based target system interface framework for UNICORE. In: 12th International Conference on ICPADS 2006, vol. 2 (2006) CD-ROM

[10] Zhang, J.: Ontology-Driven Composition and Validation of Scientific Grid Workflows in Kepler: a Case Study of Hyperspectral Image Processing. In: GCCW 2006. Fifth International Conference on Grid and Cooperative Computing Workshops, pp. 282–289 (2006)

[11] McGough, A.S., Lee, W., Darlington, J.: ICENI II. In: First International Conference on Comsware 2006, pp. 1–4 (2006)

[12] Turi, D., Missier, P., Goble, C., De Roure, D., Oinn, T.: Taverna Workflows: Syntax and Semantics. In: IEEE International Conference on e-Science and Grid Computing, Bangalore, pp. 441–448 (2007)

[13] Cao, J., Jarvis, S.A., Saini, S., Kerbyson, D.J., Nudd, G.R.: ARMS: an Agent-based Resource Management System for Grid Computing. Scientific Programming, Special Issue on Grid Computing 10(2), 135–148 (2002)

[14] Fahringer, T., Prodan, R., Duan, R., Nerieri, F., Podlipnig, S., Qin, J., Siddiqui, M., Truong, H.L., Villazon, A., ieczorek, M.: ASKALON: a Grid application development and computing environment. In: The 6th IEEE/ACM International Workshop on Grid Computing (2005) CD-ROM

[15] Fox, G., Gannon, D.: Workflow in Grid Systems. In: Concurrency and Computation: Practice & Experience, vol. 18, pp. 1009–1019. John Wiley and Sons Ltd., UK (2006)

[16] Yu, J., Buyya, R.: A Taxonomy of Workflow Management Systems for Grid Computing. Journal of Grid Computing 3, 171–200 (2005)
[17] Guan, Z., Hernandez, F., Bangalore, P., Gray, J., Skjellum, A., Velusamy, V., Liu Y.: "Grid-Flow": A Grid-Enabled Scientific Workflow System with a Petri Net-based Interface,Technical Report (December 2004), http://www.cis.uab.edu/gray/Pubs/grid-flow.pdf
[18] Sakellariou, R., Zhao, H.: A Low-Cost Rescheduling Policy for Efficient Mapping of Workflows on Grid Systems. Scientific Programming 12(4), 253–262 (2004)

An Asynchronous Parallelized and Scalable Image Resampling Algorithm with Parallel I/O

Yan Ma[1,2,3], Lingjun Zhao[1], and Dingsheng Liu[1]

[1] Center for Earth Observation and Digital Earth, Chinese Academy of Sciences(CAS)
No. 45 BeiSanHuanXi Road, P.O. Box 2434, Beijing, 100086, China
[2] Institute of Electronics, CAS. No. 19 Beisihuan Xilu, Beijing 100190, China
[3] Graduate University of Chinese Academy of Sciences
{yanma,ljzhao,dsliu}@ceode.ac.cn

Abstract. Image resampling which is frequently used in remote sensing processing procedure is a time-consuming task. Parallel computing is an effective way to speed up. However, recent parallel image resampling algorithms with massive time-consuming global processes like I/O, always lead to low efficiency and non-linear speedup ratio, especially when the amount of computing nodes increases to a certain extent. And what's more, the various geo-referencing related to different processing applications caused a real problem of code reuse. To solve these problems, Parallel Image Resampling Algorithm with Parallel I/O (PIRA-PIO) which is an asynchronous parallelized image resampling algorithm with parallel I/O is proposed in this paper. Parallel I/O of parallel file system and asynchronous parallelization which using I/O hidden policy to sufficiently overlap the computing time with I/O time are used in PIRA-PIO for performance enhancement. In addition, the design of reusable code like design pattern will be used for the improving of flexibility in different remote sensing image processing applications. Through experimental and comparative analysis, its outstanding parallel efficiency and perfect linear speedup is showed in this paper.

1 Introduction

Image resampling algorithm deals with geometric warping and transformation between two images. It plays an important role in remote sensing image processing; most of remote sensing image preprocessing procedures involve image resampling, such as Geometric Correction, Image Fusion and Image Mosaic.

Image resampling is a compute-intensive and time-consuming task. Due to the increasing spatial and spectrum resolution, the unprecedented image scales pose many computational challenges. Parallel computing with scalable cluster is an effective way to improve processing performance. Up till now, many studies on parallelization of image resampling have been probed into, including the 2D and 3D parallel image resampling algorithm on parallel machine[1], and load- balanced parallel image warping algorithm[2][3],etc. These parallel algorithms usually use Master-Slave parallel processing model. And their works mainly focus on how to reduce the computing overhead, like reducing algorithm complexity, load balancing and so on. However,

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 357–366, 2009.

there exists massive time-consuming global operation of master node in those algorithms, like serial data loading and exporting, data scattering and gathering. Moreover, with the rapidly upgrade of CPU, the performance gap between CPU and I/O is widening. As we are using a large number of CPUs (computing nodes), the I/O overhead caused by data loading and data scattering is not negligible any more, and the I/O quickly became the performance bottleneck[4].

Complete image resampling procedures commonly involve geometric mapping and interpolation. When image resampling is applied to different image processing procedures, the method of building geometric mapping varies and also the demanded auxiliary data differs, which consequently make the code reuse of this algorithm a big problem. Thus the building of different image processing applications requires developing corresponding image resampling programs which closely related to these applications, which brings another challenge to the designing and development of remote sensing image processing system.

To properly settle those issues above, PIRA-PIO an asynchronous parallelized and scalable image resampling algorithm with parallel I/O is proposed in this article. In PIRA-PIO algorithm, the data distribution and parallel I/O of parallel file system is imposed for eliminating I/O performance bottleneck, and also the I/O hidden policy which sufficiently overlaps the computing and I/O time overhead is used for further speed up. In addition, through the interface abstraction and scalability designing of algorithm with designing pattern, the image resampling program can be reused among different image processing applications.

2 Traditional Parallel Image Resampling Algorithm

Image resampling algorithm is complicated and time-consuming. In cluster environment, traditional parallel image resampling algorithms dealing with huge data generally adopt Master-Slave parallel processing model [7][8], their processing flow is illustrated as follows.

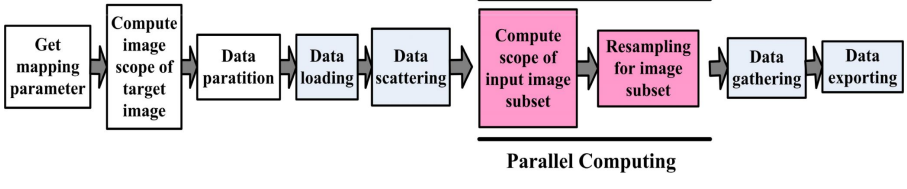


Fig. 1. Processing flow of traditional parallel image resampling algorithm

As illustrated above, the traditional parallel image resampling algorithms using classic parallel processing model which conduct a simple way of parallelization. The algorithm have good data locality only when the image distortion is small. Furthermore, it also has several problems. Firstly, it limits in performance scalability. As is showed in figure 1 that some processing steps are global operations which could not be parallelized, such as data loading, data scattering, data gathering and data exporting . While master node does data loading and data scattering, the computing nodes

are all in idle state waiting for the required data, only till the input image subset have already been fetched from master node can the resampling be continued. So, the overhead caused by resampling computing operation and I/O operation waiting for each other directly affects the parallel efficiency of whole algorithm. When a large amount of CPUs (computing nodes) are used, the problem result from the performance gap between I/O and CPU become even severe, and would finally brings perform bottleneck. Secondly, traditional parallel image resampling algorithms which closely related to applications in some aspects are not reusable.

3 PIRA-PIO Algorithm

To solve the performance problems mentioned above, we put forward PIRA-PIO algorithm an asynchronous parallelized image resampling algorithm using parallel I/O. By combining the data distribution and parallel I/O of cluster based parallel file system with the concurrent data I/O operations triggered by all computing nodes, PIRA-PIO algorithm enhances I/O performance at two layers including file system layer and algorithm layer. Moreover, based on the I/O speedup, the I/O hidden policy is also applied to overlap the I/O time with resampling computing time. Additionally, in order to improve the scalability and adaptability of this parallel algorithm, code reuse designing is carried through in algorithm building.

3.1 Algorithm Introduction

The main ideal of PIRA-PIO is: Master node responses for getting the scope of target image with mapping function, task partition according to the load measured by amount of efficient computation and task assignment. Each sub-task deals with a block of data block_t as is illustrated in figure 2. Each computing node who acts as slave node does three operations simultaneously: resampling for current block, prefetching next task and next data block of input image required for resampling, exporting the previous resampled data block of target output image. And each slave concurrently does data prefetching or exporting with parallel I/O supported by parallel file system PVFS2.

In PIRA-PIO algorithm, the master node maintains a task pool and schedules these sub-tasks to slave nodes according to a centralized load balance strategy which will probably leads to a load balance among slave nodes. And also instead of dividing and scattering the input image, computing nodes load data when needed through parallel I/O, which avoids the frequent data exchanges among computing nodes.

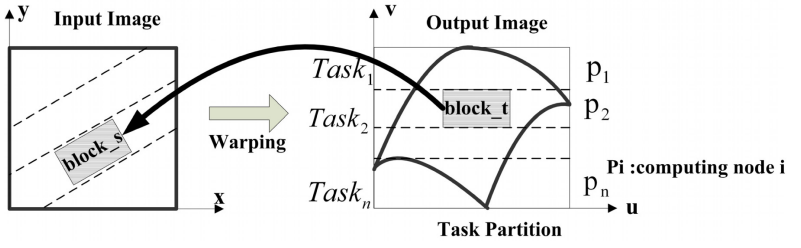


Fig. 2. Image mapping and data partition method of PIRA-PIO

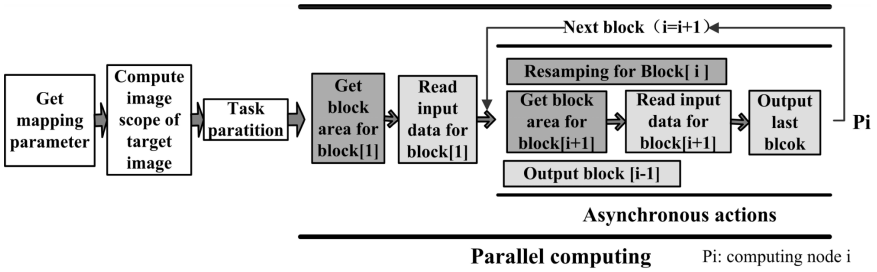


Fig. 3. Processing flow of PIWA-PIO

Further, from the flows in figure 3 we will see that there are no more data scattering and data gathering, and the data loading and exporting are implemented in parallel by each computing node concurrently. Thus the processing flow is shortened. And what's more, the data prefetching and asynchronous data I/O also lead to the sufficiently overlapped I/O time with resampling computing time.

3.2 Parallel I/O and Data Distribution

To eliminate the I/O performance, we use data distribution and parallel I/O technique in PIRA-PIO algorithm. And the remote sensing images are all stored in parallel file system PVFS2. Through PVFS2 the storage devices like disk arrays mounted to the I/O nodes in cluster are virtualized to a single file system mirror as showed in figure 4(b). The remote sensing image data in PVFS2 are partitioned into data blocks, and scattered among the disk arrays of computing nodes. The sketch map of data distribution method is showed in figure 4(a).

When implementing resampling for blocks, each computing node has to dynamically load the data in input image blocks. Due to the single file system mirror of pvfs2, all the computing nodes can access any data whenever they want. And the

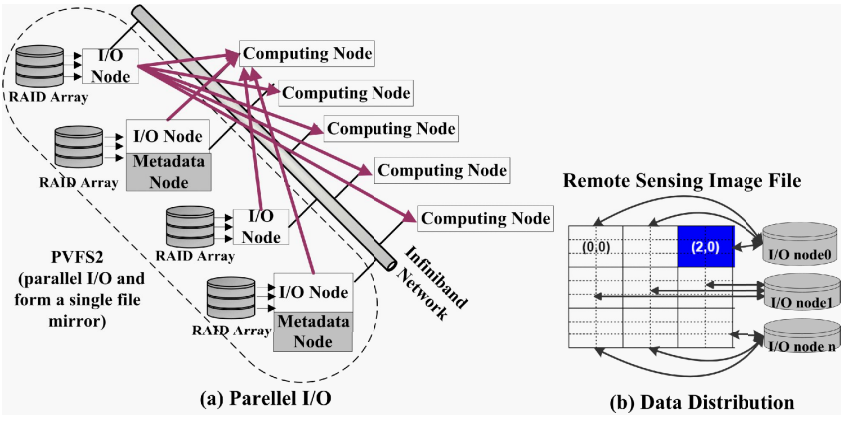


Fig. 4. Data distribution and parallel I/O

multi-stream concurrency technology also makes the concurrent data I/O operation possible. As a result, the computing nodes can concurrently read the data needed. While a data read operation is called, the parallel file system transforms it to several read actions directly performed on corresponding I/O nodes in parallel, and then the daemon process running on each I/O node fetches data from disk arrays in parallel. Also, when several data exporting or writing operations are simultaneously triggered by computing nodes after finishing data resampling, each computing node distributes the data directly to I/O nodes in parallel.

Consequently, PIRA-PIO algorithm not only supported parallel I/O in file system level, but also in algorithm level. And the parallel I/O provided at different system level significantly speedups I/O performance and allows a better scalability than ever.

3.3 I/O Hidden Strategy

Commonly, the data resampling operation has to wait while the data I/O like data loading or exporting has not yet finished, as the data used for resampling is not ready or the previous result data has not yet written back. Accordingly, even the I/O time overhead has greatly reduced by using parallel I/O, there still remains time overhead caused by the computing processing waiting for I/O. To handle the I/O waiting time issue, we put forward an I/O hidden strategy.

Data caching mechanism commonly used in the designing of storage system is an effective approach to I/O hidden. And the approach of data prefetch and asynchronous I/O used in our I/O hidden strategy is also some kind of data caching mechanism. While doing resampling for current block, it's possible for computing node to simultaneously prefetch required data for next block by calling an asynchronous I/O. Then after finishing resamping for current block, the data required for resampling next block has already prepared. Similarly, when finishes resamping for current block, the computing node would not output resampled data at once, but cache this data in buffer. Thus the computing node can start resampling for next block at once, no more wait for outputting the resampled data of current block. And that when the same time as the resamping for next block is computing, the computing can trigger an asynchronous I/O action to output the resampled data of current block that cached in buffer. As a result, data I/O operation and resampling could be implemented simultaneously, and there is no need for the resampling operation to wait for data I/O, finally the I/O could be hidden.

As is depicted in figure 5, each computing node maintains a ring buffer, which is shared among three independent threads. These three independent threads including one I/O reading thread which is responsible for getting the block area in input image for the next block and then read the data in that block area, one computing thread performs the data resampling for current block, and one I/O write thread who output the result data of previous block through pvfs2. And these three threads are putting into a pipeline as showed in following figure. When the pipeline starts, three threads work concurrently. The I/O time is mostly overlapped by computing time spent on resmapling. Finally, the I/O waiting time is almost eliminated.

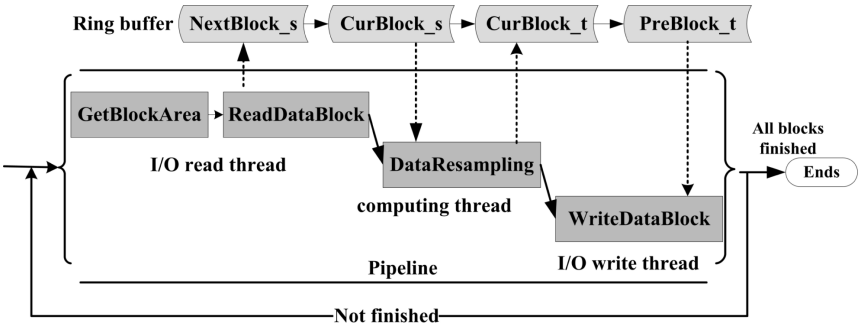


Fig. 5. I/O hidden strategy. 'NextBlock_s': input data for next block resampling, 'CurBlock_s' : input data for current block resampling, 'CurBlock_t': resampled data of current block, 'PreBlock_t': caching the resampled data of previous block.

3.4 The Code Reuse Designing

Image resampling is widely applied in most image correction processes. For example, in systematical geometric correction, a grid can be calculated by satellite orbit parameters and stance data, and then mapping relation will be obtained from the grid information, and finally follows the gray value interpolation. In ortho correction, correlative model coefficient can be calculated by satellite orbit information and ground control points, and then follows the obtaining of mapping relation with DEM, and gray value interpolation. From the traditional processing flows above, we can infer that despite of different applications, the computing flows of resampling algorithm remain almost the same. But the computing part which related to mapping relation includes obtaining methods, forms, transferring methods of it is not the same, and it is especially inconvenient for code reuse. So from the aspect of code reuse, a flexibility design based on software design pattern in algorithm implementing will be used to improve the parallel resampling algorithm's reusability in image processing system. Thereby, the programming efficiency may be improved.

When doing algorithm developing, PIRA-PIO is divided into three function modules: parallel flow control module, application computation module and gray value interpolation module. In details, parallel flow control module is in charge of realizing the parallel flow of PIRA-PIO mentioned above and making use of the correlative operation of application computation module to finish partial resampling of each computing node, grey interpolation module is responsible for calculating the appointed pixel's gray value with appointed interpolation method such as NN, BL, CC and etc, application computation module is related to applications, and it is used to compute mapping relation according to appointed criterion.

To improve the Scalability of each module, the principle of Object-Oriented design and design pattern like factory pattern and strategy pattern is used. For instance, the class diagram of application computation module is showed as Figure 6.

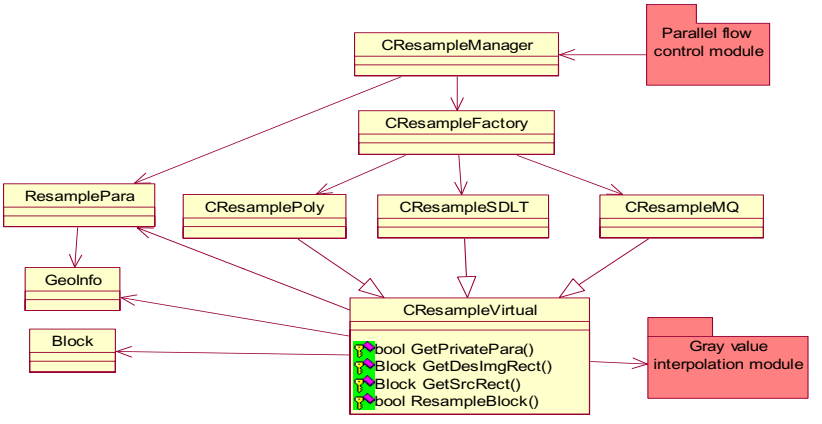


Fig. 6. The class diagram of application computation module

4 Performance Evaluation and Comparison

(1) Performance of Traditional Parallel Resampling Algorithm

The speedup ratio S_p is:

$$S_p = \frac{T_s}{T_p} \approx \frac{2T_{I/O} + T_{process}}{2T_{I/O} + 2T_{distribue} + T_{process}/N + t_{ex}} \tag{1}$$

And the data throughput ratio R is:

$$R = \frac{M}{T} = 1/\left(\frac{2}{R_{input}} + \frac{2}{N * R_{distribution}} + \frac{1}{N * R_{process}}\right) \to R_{I/O}/2 \tag{2}$$

From the above formula we can infer that when using a large amount of computing nodes, the data throughput ratio R will be tend to $0.5R_{input}$, and the $T_{I/O}$ and $T_{distribute}$ will become the limiting factor of S_p.

(2) Performance of PIRA-PIO Algorithm

Assume that the speedup ratio of the parallel I/O with N I/O nodes is K_n. If each computing node assigned m sub-tasks, then the time used for read or writing a block with parallel I/O equals $T_{I/O}/(n * m * k_n)$.

Speedup ratio S_p' is:

$$S_p' = \frac{T_s'}{T_p'} \approx \frac{2T_{I/O}/k_n + T_{process}}{2T_{I/O}/(n * m * k_n) + T_{process}/n} \to n \tag{3}$$

Data throughput ratio R ' is:

$$R' = \frac{M}{T_p'} \approx nR_{process} \tag{4}$$

As the $T_{I/O} / k_n * T_{process}$ less than 1, and $2T_{I/O} / T_{process} * m * k_n$ far less than 1, so when N is big enough, the speedup ratio S_p' will tend to N. Consequently, an elegant linear speedup ratio can be achieved. And the data through can up to N times of the data throughput ratio of the resampling processing on each computing node.

5 Experiment and Analysis

The experimental performance comparison of PIRA_PIO algorithm with traditional parallel resampling algorithm will be taken on Lenovo Deepcomp 6800 cluster. Deepcomp is composed of 18 nodes, each with dual Intel(R) 3.0Ghz processors. Network capability is 1000MBps. Three bands of Beijing-1 satellite images each with size of 15000*10000 pixels are used. And in the experiment, the image resample algorithms are applied to do a fine geometric correction with Cubic Convolution resample mode, MQ mapping model and Complete Cubic Polynomial mapping. The performance comparison is carried as follows:

(1) Comparative analysis of speedup and parallel efficiency

As is showed in the figures below that the speedup ratio and data throughput ratio of PIRA-PIO is obviously superior to the traditional one. It is proved by the experiments that the PIRA-PIO algorithm has achieved linear speedup with the increase of computing nodes. The speedup ratio is among N-1 to N, when N computing nodes are used. But for the traditional algorithm, the speedup figure is like a curve, when the amount of computing nodes is larger than four, its speedup ratio, data throughput ratio all drop sharply.

Moreover, with the increase of computing nodes, the parallel efficiency of the traditional algorithm almost descends linearly, while the parallel efficiency of PIRA-PIO always kept better than 90%. So, we can draw the conclusion that the PIRA-PIO algorithm owns an outstanding parallel efficiency and scalability in cluster platform.

(2) The analysis of the overhead of main processing steps

The main time overhead of PIRA-PIO algorithm is spent on data resampling ($T_{process}$). And the T_{io} which is spent on loading the first block and output the last result block is relatively very small. From the sub figure (b) we can see that with the increase of computing nodes, the $T_{process}$ linearly declines and T_{io} drops too. So, when enough computing nodes are used, the T_{io} can be ignored. The I/O operation almost rarely affects the performance of the whole algorithm.

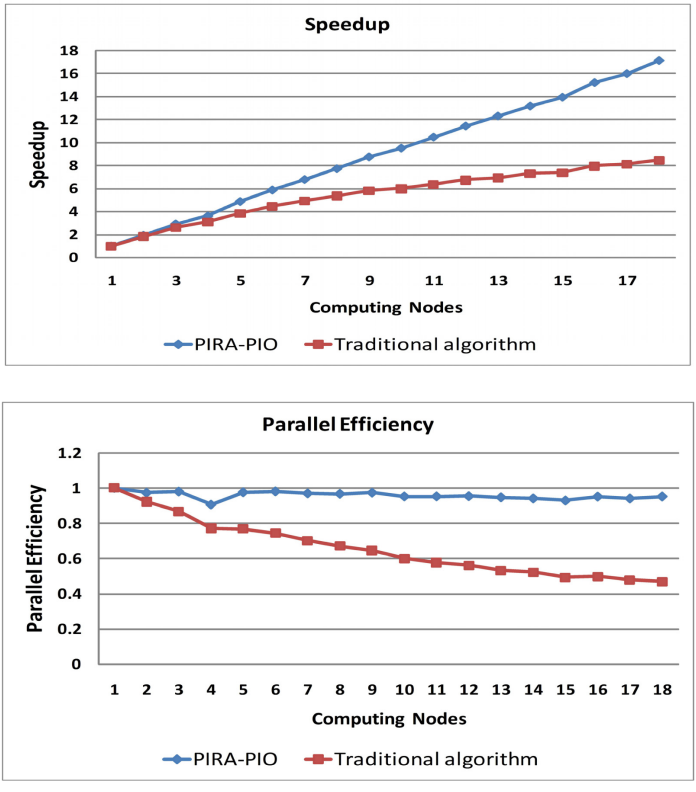
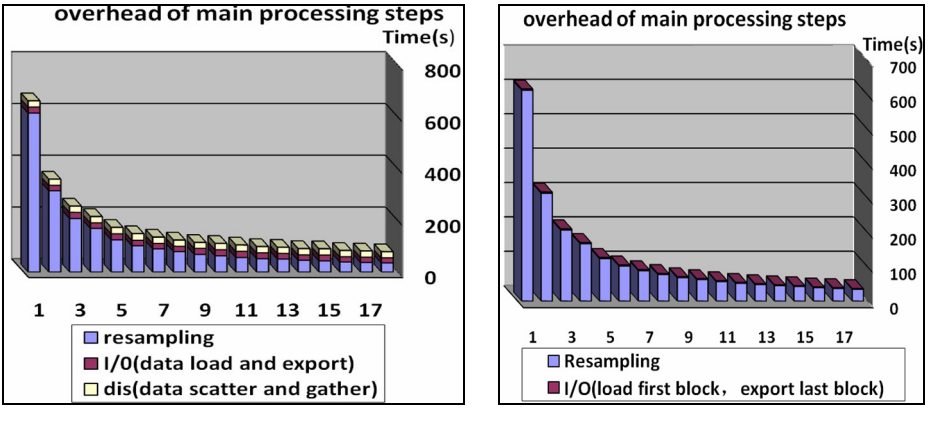


Fig. 7. Performance comparison



(a)Traditional parallel resampling algoithm (b) PIRA-PIO algoithm

Fig. 8. Time overhead of main processing steps

6 Conclusions and Future Work

The remote sensing image resampling is frequently used in various remote sensing image processing applications. It is compute-intensive, time-consuming and also poor in code reuse. In this article, to meet the performance challenge result from the I/O performance bottleneck, we propose PIRA-PIO algorithm. By adopting the data distribution and parallel I/O supported by parallel file system, using I/O hidden policy aims at eliminating I/O waiting time overhead, the performance of PIRA-PIO algorithm is significantly optimized. Through the evaluation and comparative analysis of the performance of PIRA-PIO algorithm and traditional parallel image resampling algorithm, we can see that the PIRA-PIO algorithm has gained an excellent parallel efficiency and linear speedup ratio. In addition, with the code reuse design, the flexibility of PIRA-PIO algorithm in various applications is greatly improved.

References

1. Wittenbrink, C.M., Somani, A.K.: 2D and 3D Optimal Parallel Image Warping. J. Parallel Distrib. Comput. 25(2), 197–208 (1995)
2. Contassot-Vivier, S., Miguet, S.: A load-balanced algorithm for parallel digital image warping. International journal of pattern recognition and artificial intelligence 13(4), 445–463 (1999)
3. Jiang, Y.-h., Chang, Z.-m., Yang, X.: A Load-Balanced Parallel Algo-rithm for 2D Image Warping. In: Cao, J., Yang, L.T., Guo, M., Lau, F. (eds.) ISPA 2004. LNCS, vol. 3358, pp. 735–745. Springer, Heidelberg (2004)
4. Li, G., Ma, Y., Wang, J., Liu, D.: Preliminary Through-Out Research on Parallel-Based Remote Sensing Image Processing. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3991, pp. 880–883. Springer, Heidelberg (2006)
5. Cerin, C., Jin, H.: Parallel I/O for cluster computing, London, Sterling, VA (2003)
6. Ruprecht, D., Muller, H.: Image Warping with Scattered Data. IEEE Computer Graphics and Applications 15(2), 37–43 (1995)
7. Warpenburg, M.R., Siegel, L.J.: SIMD image resampling. IEEE Transactions on Computers 31(10), 934–942 (1982)
8. Contassot-Vivier, S., Miguet, S.: A load-balanced algorithm for parallel digital image warping. International journal of pattern recognition and artificial intelligence 13(4), 445–463 (1999)
9. Lingjun, Z., Dingshen, L., Guoqing, L., wenyi, Z.: A Study Of High-Performance Precision Correction For Satellite Image. Remote Sensing For Land & Resources (1), 49–52 (2007)
10. Zhaoxia, F., Yan, H., Bo, Z.: An Automatic and Robust Image Mosaic Algorithm. Telecommunication Engineering 47(3), 55–58 (2007)
11. Webb, R.D.: Object Description Language [P]. United States Patent Application: 20030070159 (2003)

Design and Implementation of a Scalable General High Performance Remote Sensing Satellite Ground Processing System on Performance and Function

Jingshan Li and Dingsheng Liu

Center for Earth Observation and Digital Earth, Chinese Academy of Sciences,
No.45, Bei San Huan Xi Road, Beijing, China
{jsli,dsliu}@ceode.ac.cn

Abstract. This paper discusses design and implementation of a scalable high performance remote sensing satellite ground processing system using a variety of advanced hardware and software application technology on performance and function. These advanced technologies include the network, parallel file system, parallel programming, job schedule, workflow management, design patterns,etc, which make performance and function of remote sensing satellite ground processing system scalable enough to fully meet the high performance processing requirement of multi-satellite, multi-tasking, massive remote sensing satellite data. The "beijing-1" satellite remote sensing ground processing system is introduced as an instance.

Keywords: parallel computing, remote sensing, data processing system, clusters computing, scalable.

1 Introduction

With development of the remote sensing mini-satellites, mini-satellites constellation, as well as the high-performance parallel computing technology, the remote sensing satellite ground processing system faces to the demand of multi-platform, multi-tasking, high performance data processing. Remote sensing is characterized by a need to perform computationally intensive operations on large data sets. Full-scene image processing requires operating on tens of millions of image data points per scene.

The cluster of PC server has grown from a curiosity to become the norm for much of the world's computing [1],[5],[6].Integrating parallel computer programs into a framework that can be easily used by applied scientists is a challenging problem. Such a framework has to enable simplified access to computationally complex operations and high performance technologies, as well as providing a means for defining the appropriate data sets for the operation request. It still can be inadequate for a lack of suitable software, suitable hardware and total integrate processing system [2], [3],[4].

The scalable General HIgh Performance remote sensing satellite ground Processing System (GHIPS) is researched. The advanced high performance computing technology, advanced software design, a combination of parallel storage technologies, parallel computing, task scheduling technology, process management technology, design

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 367–374, 2009.

methods, web service technology are used in this system, which makes performance and function to achieve a high degree to meet the requirement of satellite ground data processing system.

We describe our GHIPS design on performance and function in section 2 and GHIPS implementation case, the "beijing-1" satellite remote sensing ground processing system, in section 3.We discuss conclusion in section 4.

2 GHIPS Design on Performance and Function

GHIPS, developed at the Center for Earth Observation and Digital Earth, Chinese Academy of Sciences, is a high performance computing middleware that supports the development and execution of generic remote sensing satellite ground processing system applications over a collection of hardware and software resources. The system is client-server architecture. The client submitted to the task of processing through the web service protocol to server end, and server end finished product processing with high performance.

This paper focuses on the server-side design of the system. The key issue of remote sensing satellite ground processing system on performance and function is researched.

As shown in Figure 1, the server end system is divided into five layers, namely the hardware layer, system support layer, data processing layer, tasks scheduling layer and user composition layer. The first two layers make system scalable on performance and the other three layers make system scalable on function. The methods of making system scalable on performance and function are discussed below.

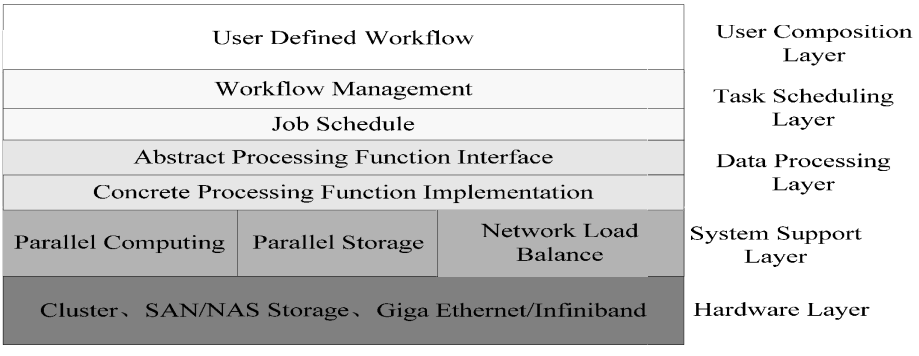


Fig. 1. GHIPS Server End Architecture

2.1 Hardware Layer

The hardware layer provides hardware support, including commercial off-the-shelf systems such as PC server cluster, Storage Area Network (SAN) / Network Attached Storage (NAS) storage systems, Gigabit Ethernet or Infiniband network systems.

All three modes use traditional network storage technologies: direct attached storage (DAS), which is attached to a single server; NAS, a server dedicated to file sharing; and SAN, networks of shared storage devices. NAS can offer low cost massive

storage, and SAN can offers both high performance network storage and secure data sharing across platforms.

The InfiniBand is a switched fabric communications link primarily used in high-performance computing. Its features include quality of service and failover, and it is designed to be scalable. The InfiniBand architecture specification defines a connection between processor nodes and high performance I/O nodes such as storage devices. The InfiniBand can offer high-bandwidth, low-latency communication among cluster. The InfiniBand standard supports single data rate (SDR), double data rate (DDR), and quad data rate (QDR),which provides bandwidth equal to 10Gbps, 20Gbps,and 40Gbps, respectively.

The commercial off-the-shelf computing hardware is a cost-effective way of exploiting in remote sensing applications. The above high performance hardwares have standard interface and make system scalable on performance of computing, storage and network. On this hardware layer, users can deployment commercial off-the-shelf software without worrying about any further change of hardware.

2.2 System Support Layer

The remote sensing data processing is a complex computing intensive, data intensive and network intensive applications and different functions of computing, storage and network access performance requirements may differ from each other.

The system support layer provides software support environment, includes parallel storage software, such as Stornext, PVFS, Luster file system, parallel computing software, such as MPI, OpenMP, PVM, network load balance software, such as LVS, BigIP. This layer can integrate and make full use of computing, storage, and network access software and resource. The orthogonal design of the system of parallel computing, parallel storage, the parallel network load make the system performance, network performance, storage performance to be scalable independent, and to be optimized data processing performance on the three dimensions. The optimized method of three dimensions is showed in Figure 2.This layer shields the complexity of using of parallel software, and users can design and implement their data processing application without worrying about any further change of parallel system support software, and can gain the better performance of new system software and hardware automatically.

2.3 Data Processing Layer

The data processing layer makes use of advanced software design patterns. In software engineering, a design pattern is a general reusable solution to a commonly occurring problem in software design. A design pattern is not a finished design that can be transformed directly into code. It is a description or template for how to solve a problem that can be used in many different situations. Object-oriented design patterns typically show relationships and interactions between classes or objects, without specifying the final application classes or objects that are involved.

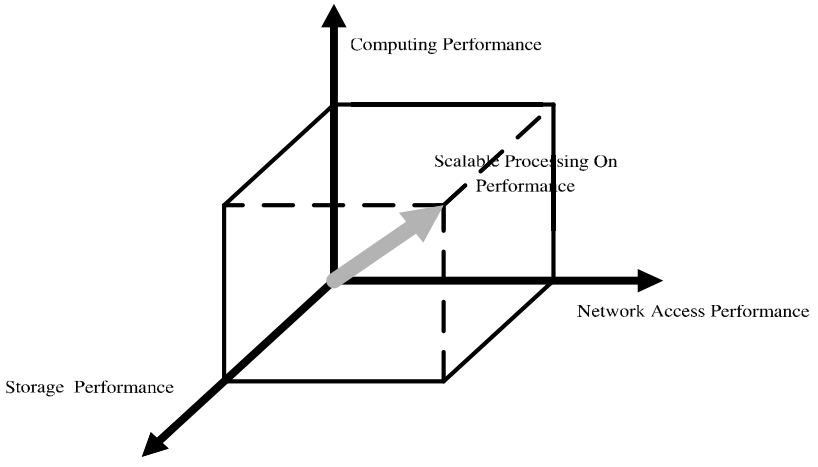


Fig. 2. Scalable on Performance

A software design pattern, the abstract factory Pattern provides a way to encapsulate a group of individual factories that have a common theme. In normal usage, the client software would create a concrete implementation of the abstract factory and then use the generic interfaces to create the concrete objects that are part of the theme. The client does not know (or care) about which concrete objects it gets from each of these internal factories since it uses only the generic interfaces of their products. This pattern separates the details of implementation of a set of objects from its general usage.

The template design pattern is perhaps one of the most widely used and useful design patterns. It is used to set up the outline or skeleton of an algorithm, leaving the details to specific implementations later. This way, subclasses can override parts of the algorithm without changing its overall structure. This is particularly useful for separating the variant and the invariant behavior, minimizing the amount of code to be written. The invariant behavior is placed in the abstract class (template) and then any subclasses that inherit it can override the abstract methods and implement the specifics needed in that context.

Using design pattern, data processing layer design a variety of abstract function interface for a variety of satellite sensors using abstract factory design pattern and common logical thread of public functions using template method design pattern. The various processing algorithms of satellite sensors can be added through inherit, which ensure scalability of satellite sensors data processing algorithms. The scalable data processing layer architecture showed as class diagram is illustrated in Figure 3, for examples of data processing of "beijing-1" satellite and " CBERS " satellite.

2.4 Task Scheduling Layer

A cluster system has driven the need for new job scheduling system in the computational science realm. A job scheduling system is used for both serial and parallel job control.

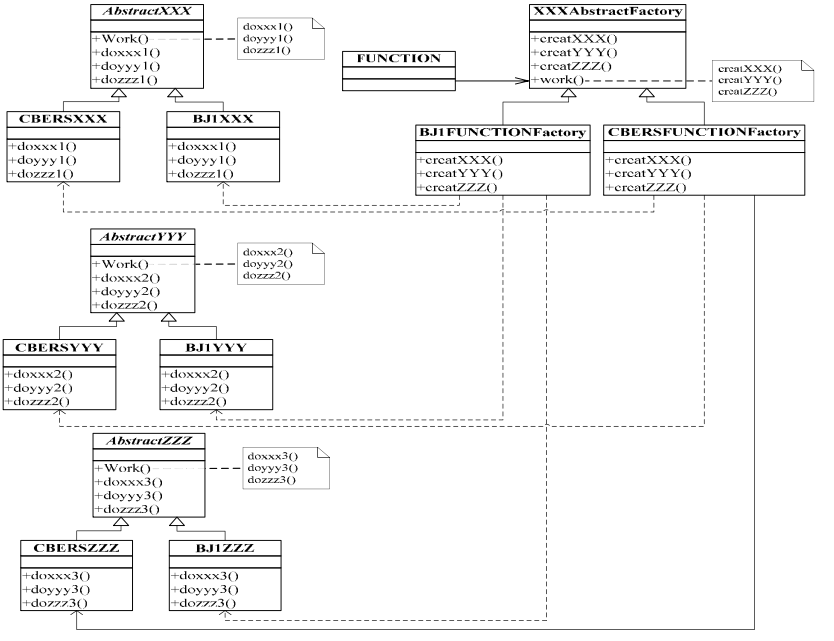


Fig. 3. Scalable Data Processing Layer Architecture and Class Diagram

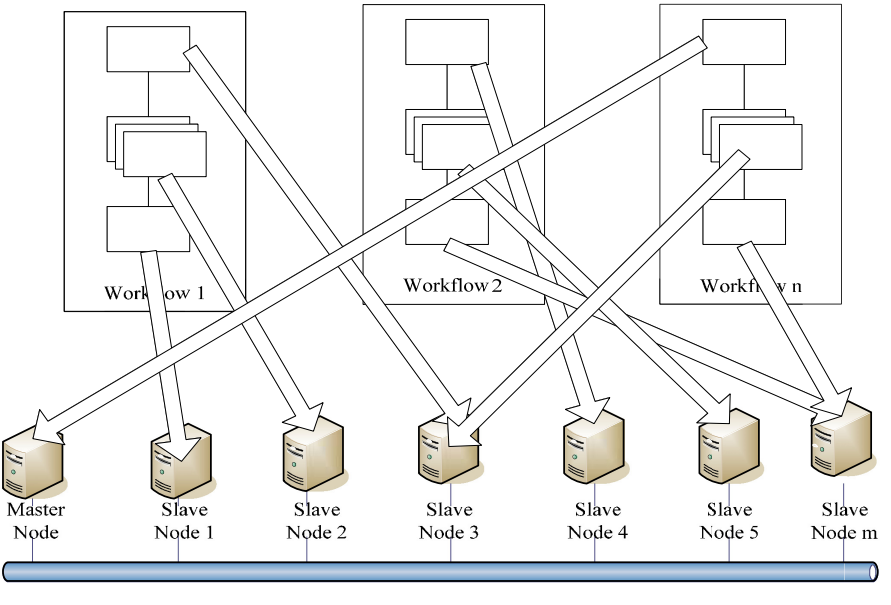


Fig. 4. Tasks Scheduling System

A workflow is a depiction of a sequence of operations, declared as work of a person, work of a simple or complex mechanism, work of a group of persons, work of an organization of staff, or machines. Workflow software can provide end users with an easier way to orchestrate or describe complex processing.

The tasks scheduling layer integrate job scheduling and workflow management system to achieve the work load balancing process and the ability to compose complex job workflow. The tasks scheduling layer can assign any data processing job on the whole cluster with help of system support software. The tasks schedule is illustrated in Figure 4.

2.5 User Composition Layer

The user composition layer provides users with kind interface to compose process, users can compose a variety of complex processing, such as data processing, archiving, production, distribution and other functions, as shown in Figure 5.

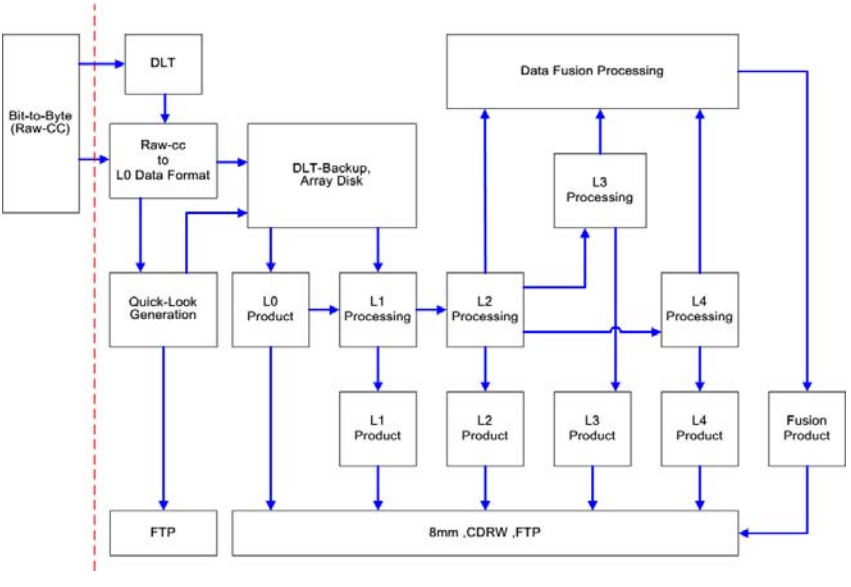


Fig. 5. User Composition Data Processing Task

2.6 Scalable on Performance and Function

The hardware layer and system support layer achieve on demand scalability on system performance. The using of high performance commercial off-the-shelf computing hardware and orthogonal design of the system of parallel computing, parallel storage, and the parallel network load make the system performance, network performance, storage performance to be scalable independent, and to be optimized data processing performance on the three dimensions.

The data processing layer, task scheduling layer and user composition layer achieve on demand scalability on system function, that not only to increase the number of new satellite remote sensing data processing algorithms and functions, but also

bring out a variety of composition of the basic processing functions. Users can customize remote sensing data processing to meet user custom needs of data processing.

The system has a completely open platform due to the use of a variety of commercial off-the-shelf advanced high performance computing technology and advanced software design technology. Users can facilitate the balanced expansion of the system of processing, storage and network capacity to improve the performance of the process and load their own business functions of a new satellite remote sensing data by inserting the appropriate data processing module based on certain rules to extend system processing function without modifying the system hardware and software.

3 GHIPS Implementation Case

Based on above research, The new version of "Beijing-1" remote sensing satellite ground processing system is implemented on GHIPS, which make use of NAS storage systems, Gigabit Ethernet, and computing servers hardwares, PVFS2,Parallel Virtual File System Version 2,MPICH2,an implementation of the Message-Passing Interface, openPBS, open source version of the Portable Batch System, OSworkflow, a pure Java open-source workflow engine for technical users, and JAVA RMI, Remote Method Invocation.

The system include system geometric correction, geometric correction based on control point and Digital Elevation Model (DEM), and other functions more than a dozen modules of serial and parallel processing, more than 30 kinds of business of composition of modules, and more business can be composed in accordance with the requirements of users.

The cluster consists of 16 SMP PC computing nodes connected by 24 port gigabit Ethernet switch. The computing node has two Intel Xeon 2.4GHz CPU and 1G Bytes local memory. The eight nodes are used for new version based on GHIPS; the other eight nodes are used for previous version of "Beijing-1" remote sensing satellite ground processing system.

The test data processing includes raw data management, band registration, MTF correction. The system shows about 20% performance improvement over the previous version with using the local file system and custom scheduling system on the same hardware. The test data and result are showed in table 1.

Table 1. New version based on GHIPS versus previous version

Dataset Name	Data Row Number	time-consuming(second)	
		New version	Previous version
dmc+4bj1L104081701 60041_160155_0	5000~14983	777	851
dmc+4bj1L104081880 32203_032703_0	10000~19983	711	896
dmc+4bj1L104081880 32203_032703_0	50000~59983	718	931
dmc+4bj1L104081910 20724_020844_0	1~9984	654	903

4 Conclusions

As many remote sensing satellite ground processing applications for computing have high demand for processing powers, it is critical to have a platform that readily supply such computing powers with the ease of use.

In this paper we propose the architecture of GHIPS. With using of a variety of commercial off-the-shelf advanced high performance computing technology and advanced software design technology, users can facilitate the balanced expansion of the system of processing, storage and network capacity to improve the performance of the process and load their own business functions of a new satellite remote sensing data by inserting the appropriate data processing module based on certain rules to extend system processing function without modifying the system hardware and software.

Future work is divided between application issues and infrastructure issues. The application issues include the integration new satellite remote sensing data processing and basic algorithms. The infrastructure issues are studying how parallelization at various hardware and software levels can be used for maximization of performance and researching corresponding algorithms parallel model.

References

1. Chao-Tung, Y., Chih-Li, C., Chi-Chu, H., Frank, W.: Using a BEOWULF cluster for a remote sensing application. In: Proceedings of the 22nd Asian Conference on Remote Sensing, Singapore, pp. 233–238 (2001)
2. Li, G., Liu, D.: Key Technologies Research on Building a Cluster-Based Parallel Computing System for Remote Sensing. In: International Conference on Computational Science, vol. (3), pp. 484–491 (2005)
3. Zhang, W., Liu, D., Li, G., Zhang, W.: Special Task Scheduling and Control of Cluster Parallel Computing for High-Performance Ground Processing System. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3993, pp. 17–23. Springer, Heidelberg (2006)
4. Hawick, K.A., Maciunas, K.J., Vaughan, F.A.: DISCWorld: A Distributed Information Systems Control System for High Performance Computing Resources. DHPC Technical Report DHPC-014, Department of Computer Science, University of Adelaide (1997)
5. Yang, C.T.: Using a Beowulf Cluster for a Remote Sensing Application. In: Proceedings of the Asian Conference on Remote Sensing (ACRS), Singapore (2001)
6. Sterling, T.L., Salmon, J., Backer, D.J., Savarese, D.F.: How to Build a Beowulf: A Guide to the Implementation and Application of PC Clusters 2nd Printing. MIT Press, Cambridge (1999)

Incremental Clustering Algorithm for Earth Science Data Mining

Ranga Raju Vatsavai

Computational Sciences and Engineering Division
Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA
vatsavairr@ornl.gov

Abstract. Remote sensing data plays a key role in understanding the complex geographic phenomena. Clustering is a useful tool in discovering interesting patterns and structures within the multivariate geospatial data. One of the key issues in clustering is the specification of appropriate number of clusters, which is not obvious in many practical situations. In this paper we provide an extension of G-means algorithm which automatically learns the number of clusters present in the data and avoids over estimation of the number of clusters. Experimental evaluation on simulated and remotely sensed image data shows the effectiveness of our algorithm.

Keywords: Clustering, EM, GMM, Remote Sensing, K-means, G-means.

1 Introduction

Remote sensing, which provides inexpensive, synoptic-scale data with multi-temporal coverage, has proven to be very useful in land cover mapping, environmental monitoring, forest and crop inventory, urban studies, natural and man made object recognition, etc. Thematic information extracted from remote sensing imagery is also useful in a variety spatio-temporal applications. For example, land management organizations and the public have a need for more current regional land cover information to manage resources and monitor land use changes. Likewise, intelligence agencies, such as, National Geospatial Intelligence Agency (NGA), and Department of Homeland Security (DHS), utilizes pattern recognition and data mining techniques to classify both natural and man made objects from large volumes of high resolution imagery.

Clustering algorithms play a key role in earth science data mining. They are often used to analyze complex and large volumes of multivariate geospatial data, such as, remotely sensed images, sensor measurements, field observations, etc., as a first step in gaining insights into the structure or natural groupings. Clustering is also used to in compression, exploratory analysis, and summarization of the data. Cluster analysis is used in many other spatial and spatiotemporal application domains. Cluster analysis is routinely used in epidemiology for finding unusual groups of health-related events. Cluster analysis is also used in detection of crime hot spots.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 375–384, 2009.

One of the key challenges in clustering is the specification of the number of clusters. Determining an optimal number of clusters manually is not feasible given the complexity and volume of geospatial data sets. In this paper we provide a simple extension of the G-means [3] algorithm that automatically discovers the number of clusters. Experimental evaluation shows that our algorithm can avoid the common problem of ending up with a large number of spurious clusters by the G-means algorithm.

2 Related Work and Our Contributions

Clustering is a fertile research area with applications cutting across many domains. A large number of clustering algorithms can be found in the literature [4,5]. These algorithms can be broadly categorized into: hierarchical, partitional, density-based, and grid-based methods. Partitional clustering algorithms, especially, K-Means algorithm is very popular in several application domains, including earth sciences. One of the key inputs to K-Means algorithm is the specification of K, the number of clusters. However, determining an optimal number of clusters manually is not feasible given the complexity and volume of geospatial data sets.

Considerable research has gone into finding the optimum number of clusters directly from the data itself [2,6,7,8,13]. In [1,3] authors proposed a G-means algorithm that automatically discovers the number of clusters. Basic idea behind G-means is simple. Initial number of clusters (k) determined by k-means are incremented by splitting each cluster that doesn't pass a statistical test. The clustering process is repeated until all the clusters have passed this statistical test. In many practical situations there is a danger of over estimating the number of clusters, especially if the model is assumed to be a Gaussian Mixture Model (GMM). We extended the G-means algorithm to overcome this practical limitation. In many situations G-means clustering algorithm tends to find more clusters. In order to reduce the chance of finding more clusters, we devised a new approach that prevents some splits and allows to reverse the splits. In a nutshell we made two modifications to the G-means algorithm. First, instead of univariate test statistic, we used a multivariate test statistic, known as, Shapiro-Wilk statistic. This modification has following advantages. First we don't have to project multivariate data into 1-d. This is important for earth science data mining as the geospatial data is often high-dimensional in nature. Finding a good projection can be as difficult as finding a good K. Second, AD test is good for small samples, that is, number of samples ≤ 25. However, in earth science data sets typically we have large number of samples (per cluster). Finally, the multivariate Shapiro-Wilk test exhibits good power against alternatives [10]. We used KL Divergence measure after splitting the clusters to see if any pair of clusters are too close to each other. If any two clusters are too close to each other, then it is better to combine them, even though such combination may violate significance testing. In the following sections, we present our algorithm and experimental results.

3 Clustering Framework

Basic statistical framework for our clustering approach is Gaussian Mixture Models (GMMs). Typically model based clustering approaches are not applied on entire data set given the computational and data complexity. Rather a subset of data samples are collected from the full data set. Model parameters are estimated using these samples. Once a model is constructed, all the samples (data points) in the full data set can then be assigned to one of the clusters, based on some distance (or decision) criteria. Our algorithm is based on the assumption that the data samples are generated by a GMM. Then the objective is to learn the GMM parameters from these samples. We now briefly describe an expectation maximization based algorithm to learn the GMM parameters.

3.1 Estimating GMM Parameters

Let us now assume that the sample data set $D = \{x_i\}_{i=1}^{n}$ is generated by the following mixture density.

$$p(x_i|\Theta) = \sum_{j=1}^{K} \alpha_j p_j(x_i|\theta_j) \tag{1}$$

Here $p_j(x_i|\theta_j)$ is the pdf corresponding to the mixture j and parameterized by θ_j, and $\Theta = (\alpha_1, \ldots, \theta_K, \theta_1, \ldots, \theta_K)$ denotes all unknown parameters associated with the K-component mixture density. For a multivariate normal distribution (eq. 2), θ_j consists of elements of the mean vectors μ_j and the distinct components of the covariance matrix Σ_j.

$$p(x|y_j) \;=\; \frac{1}{\sqrt{(2\pi)^{-N}|\Sigma_j|}} e^{\frac{-1}{2}(x-\mu_j)^t|\Sigma_j|^{-1}(x-\mu_j)} \tag{2}$$

The *log-likelihood* function for this mixture density can be defined as:

$$L(\Theta) = \sum_{i=1}^{n} \ln \left[\sum_{j=1}^{M} \alpha_j p_j(x_i|\theta_j) \right]. \tag{3}$$

In general, Equation 3 is difficult to optimize because it contains the ln of a sum term. However, this equation greatly simplifies in the presence of unobserved (or incomplete) samples. Typically, we assume that the cluster labels as missing (unobserved) data, and use expectation maximization technique to estimate parameters (Θ). The EM algorithm consists of two steps, called the E-step and and M-step as given below.

E-Step. For multivariate normal distribution, the expectation $E[.]$, which is denoted by p_{ij}, is the probability that Gaussian mixture j generated the data point i, and is given by:

$$p_{ij} = \frac{\left|\hat{\Sigma}_j\right|^{-1/2} e^{\left\{-\frac{1}{2}(x_i-\hat{\mu}_j)^t \hat{\Sigma}_j^{-1}(x_i-\hat{\mu}_j)\right\}}}{\sum_{l=1}^{M} \left|\hat{\Sigma}_l\right|^{-1/2} e^{\left\{-\frac{1}{2}(x_i-\hat{\mu}_l)^t \hat{\Sigma}_l^{-1}(x_i-\hat{\mu}_l)\right\}}} \tag{4}$$

Table 1. Algorithm for Computing Parameter of Finite Gaussian Mixture Model Over Unlabeled Training Data

Inputs: D, sample data set; K, the number of clusters.
Initial Estimates: Do clustering by K-Means, and estimate initial parameter using Maximum Likelihood Estimation (MLE) technique to find $\hat{\theta}$.
Loop: While the complete data *log-likelihood* improves:
E-step: Use current classifier to estimate the class membership of each unlabeled sample, i.e., the probability that each Gaussian mixture component generated the given sample point, p_{ij} (see Equation 4).
M-step: Re-estimate the parameter, $\hat{\theta}$, given the estimated Gaussian mixture component membership of each unlabeled sample (see Equations 5, 6, 7)
Output: Parameter vector Θ.

M-Step. The new estimates (at the k^{th} iteration) of the model parameters in terms of the old parameters are computed using the following update equations:

$$\hat{\alpha}_j^k = \frac{1}{n} \sum_{i=1}^{n} p_{ij} \tag{5}$$

$$\hat{\mu}_j^k = \frac{\sum_{i=1}^{n} x_i p_{ij}}{\sum_{i=1}^{n} p_{ij}} \tag{6}$$

$$\hat{\Sigma}_j^k = \frac{\sum_{i=1}^{n} p_{ij}(x_i - \hat{\mu}_j^k)(x_i - \hat{\mu}_j^k)^t}{\sum_{i=1}^{n} p_{ij}} \tag{7}$$

The EM algorithm iterates over these two steps until convergence is reached. We can now put together these individual pieces into the following algorithm (Table 1) which computes the parameters for each component in the finite Gaussian mixture model that generated our sample data D (without any cluster labels).

3.2 Simulation Example 1

We now demonstrate GMM clustering algorithm (1) on a simulated data set. We generated a GMM with three components. The parameters are given in Table 2 and Table 3. We generated 150 bivariate Gaussian samples from each

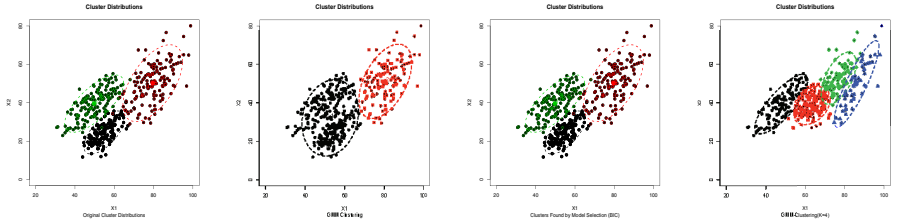
Table 2. Simulation Parameters (Mean)

	x	y
C1	55.00	25.00
C2	80.00	50.00
C3	50.00	40.00

Table 3. Simulation Parameters (Covariance)

	C1		C2		C3	
	x	y	x	y	x	y
x	30.00	25.00	60.00	40.00	60.00	50.00
y	25.00	40.00	40.00	90.00	50.00	70.00

component density. We applied GMM clustering algorithm on this sample data set by assuming different K values and the results were summarized in Figure 1. From the figure it can be seen that when K assumption is correct (that is, K=3) we have very good estimates (compare subfigures (a) and (c)), however for other K's (subfigures (b) and (d)) the estimates are very different than the original distribution. This simulation emphasizes the need to estimate a good K value (if possible automatically from the data).



(a) Simulated (K=3) (b) Estimated (K=2) (c) Estimated (K=3) (d) Estimated (K=4)

Fig. 1. Simulated vs. Estimated (GMM-Clustering for different K values)

4 Learning to Estimate K

In this section we address the problem of estimating K automatically from the data. As with the estimation of the model parameters for finite Gaussian mixture model, we assume that the training dataset D is generated by a finite Gaussian mixture model, but we don't know either the number of components or the labels for any of the mixture component. In the previous section, we devised an algorithm to find parameters by assuming a K-component finite Gaussian mixture model. In general, we can estimate parameters for any arbitrary K-component model, as long as there are sufficient number of samples available for each component and the covariance matrix does not become singular. Then the question remains, which K-component model is better? This question is addressed in the area of model selection, where the objective is to chose a model that maximizes a cost function. There are several cost functions available in the literature, most commonly used measures are Akaike's information criterion (AIC), Bayesian information criteria (BIC), and minimum description length (MDL). The common criteria behind these models is to penalize the models with additional parameters, so BIC and AIC based model selection criteria follows the principal of parsimony. In this study we considered BIC as a model selection criteria, which also takes the same form as MDL. We also chose BIC, as it found to be very useful in model based clustering [2], and also because it is defined in terms of maximized log-likelihood which any way we are computing in our parameter estimation procedure defined in the previous section. BIC can be defined as

$$BIC = MDL = -2 \log L(\Theta) + m \log(N) \tag{8}$$

where N is the number of samples and m is the number of parameters. We now describe our BIC based model selection criteria to determine the number

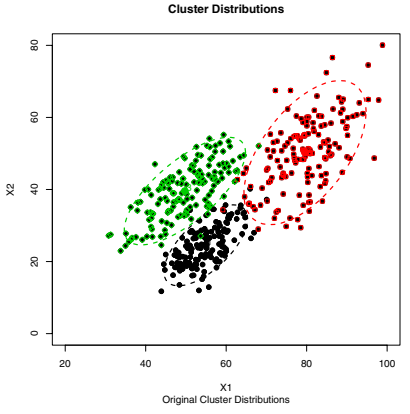
components in each aggregate class. First, we take the aggregate class and split it into two Gaussians at a time using the Gaussian splitting criteria specified in [11]. Then the parameters of this new mixture model are estimated using the algorithm 1. This process is recursively applied for a fixed number times or BIC is minimized.

On the other hand, G-means [3] clustering is initialized by k-means clustering for suitable initial K value. Each cluster is then tested for normality using univariate test: Anderson-Darling (AD) statistic. For a user given p-value, if AD test fails, then the cluster is split into two clusters. K-means cluster is performed again with new K, and the process is repeated until no more (new splits) clusters can be found. Multivariate data is projected on to 1-d to facilitate AD test. In [3], authors argued that BIC has a tendency to find more clusters. In our experiments, we found that G-means clustering also tend to find more clusters. We demonstrate this through an example simulated data set. In order to reduce the chance of finding more clusters, we devised a new approach that prevents splits and allows to reverse the splits.

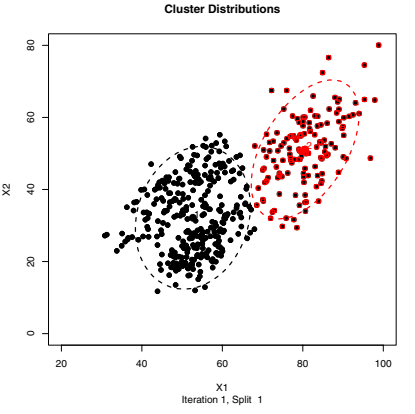
Our algorithm differs from G-means in two ways. First, instead of univariate test statistic, we used a multivariate test statistic, known as, Shapiro-Wilk statistic. More details on Shapiro-Wilk test can be found in [12]. This modification has following advantages. First we don't have to project multivariate data into 1-d. This is important for earth science data mining as the geospatial data is often high-dimensional in nature. Finding a good projection can be as difficult as finding a good K. Second, AD test is good for small samples, that is, number of samples ≤ 25. Finally, the multivariate Shapiro-Wilk test exhibits good power against alternatives [10]. Finally, statistical tests are sensitive to noise. It is likely that splitting process (increasing K) continue beyond optimal K as many times statistical significance test fails (even though clusters are close to multivariate normal). As a check to prevent this happening, we added additional criteria to check for the quality of splits. We used KL Divergence [9] measure after splitting to see if any pair of clusters are too close to each other. If any two clusters are too close to each other, then it is better to combine them. The new algorithm (GMM-Adaptive-K) is summarized in Table 4.

4.1 Simulation Example 2

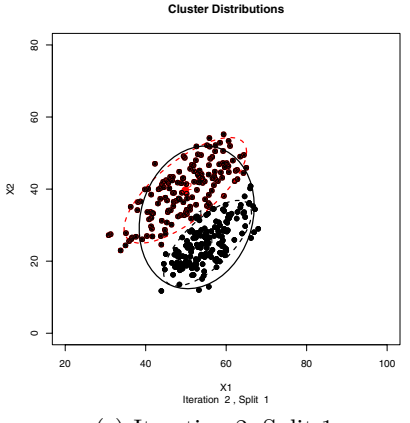
We now demonstrate GMM-Adaptive-K clustering algorithm (4) on the simulated data set (Table 3). The results were summarized in Figure 2. First iteration found two clusters (Figure 2(b)), red cluster passes Shapiro-Wilk test. As result, only the 2nd cluster (black) is split into two clusters (c). In the next iteration, red cluster failed Shapiro-Wilk test, as a result it was split into two clusters (d). The G-means cluster algorithm would have resulted in a final solution shown in Figure 2(e). On the other-hand the additional step introduced in our algorithm, finds that these two clusters are very close (KL-Divergence), thus decrements number of clusters to 3. Final solutions is shown in Figure 2(f). Compare Figure 2(f) with original distribution in Figure 2(a).



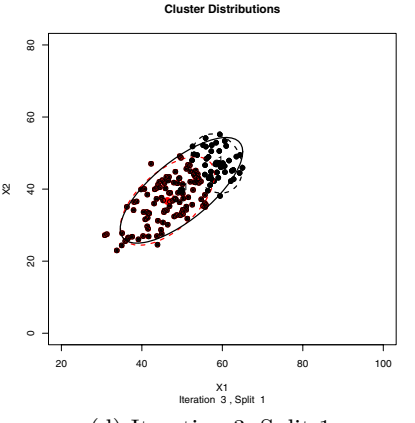
(a) Original Distribution



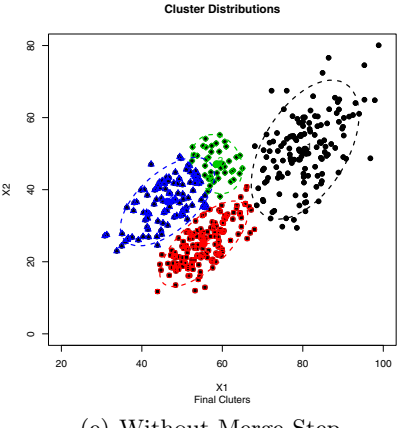
(b) Iteration 1, Split 1



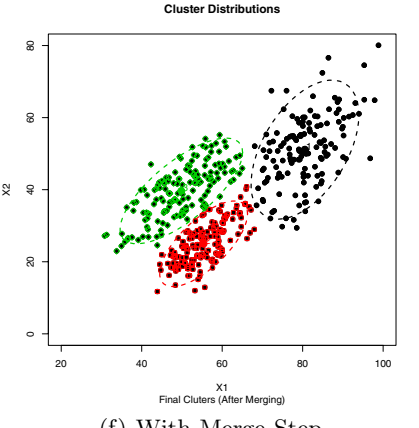
(c) Iteration 2, Split 1



(d) Iteration 3, Split 1



(e) Without Merge Step



(f) With Merge Step

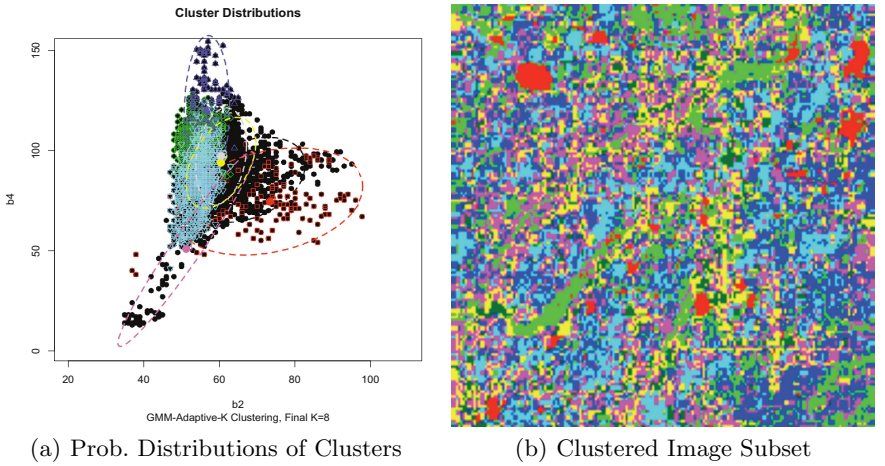
Fig. 2. GMM-Apdaptive-K Algorithm Trace

Table 4. GMM-Adaptive-K Algorithm

Inputs: D, sample data set; significance (default p-value = 0.05), initial K (default = 2), nClusters = K
Clustering: GMM-Clustering (see Algorithm 1)
Loop 1: WHILE (TRUE):
Loop 2: FOR 1:nClusters
Statistical test: Shapiro-Wilk test.
Check: IF a cluster fails statistical test, split that cluster into two clusters using GMM-Clustering; increment nClusters and K; ELSE accept cluster, decrement nClusters
Clustering: GMM-Clustering(failed-cluster-data-samples, new K)
Merge: Compute KL-Divergence, IF two-clusters are closer than threshold value, decrement K, continue (Loop 2)
Check: IF nClusters = 0 (break, Loop 1)
Output: Parameter vector Θ.

5 Experimental Results

We have applied our GMM-Adaptive-K algorithm on the real data set described below.The Cloquet study site encompasses Carlton County, Minnesota, which is approximately 20 miles southwest of Duluth, Minnesota. The region is predominantly forested, composed mostly of upland hardwoods and lowland conifers. There is a scattering of agriculture throughout. The topography is relatively flat, with the exception of the eastern portion of the county containing the St. Louis River. Wetlands, both forested and non-forested, are common throughout the area. The largest city in the area is Cloquet, a town of about 10,000. We used a spring Landsat 7 scene, taken May 31, 2000, and clipped to the study region. The final rectified and clipped image size is 1343 lines x 2019 columns x



(a) Prob. Distributions of Clusters (b) Clustered Image Subset

Fig. 3. GMM-Apdaptive-K Algorithm on Carleton Satellite Image Data

6 bands. We selected 400 random plots. From each plot, we extracted 9 feature vectors (6 dimensional) by placing a 3 x 3 window at the center of each plot. That is, sample data set consisted of of 3600 feature vectors. We applied our GMM-Adaptive-K algorithm on this sample data set and found 8 clusters. In a supervised classification experiment, remote sensing analysts have identified 10 classes for this study site. Supervised classification image was visually compared with our clustering algorithm. It appears a good correspondence between the clusters and thematic classes identified by the analyst. However, further analysis and experimentation is needed to establish this correspondence between the clusters and thematic (information) clusters. Figure 3(a) shows the cluster (bivariate density) distributions in feature space (bands 2 and 4), and Figure 3(b) shows a small clip from the clustered image.

6 Conclusions

We developed an incremental clustering algorithm. The algorithm is based on GMM distribution and expectation maximization (EM) parameter estimation. The algorithm is also an extension of G-means algorithm, which splits clusters failing statistical significance tests, in a iterative manner to find optimal number of clusters. However, our algorithm avoids an important limitation of over estimation of K by employing KL divergence measure to find highly overlapping clusters and try to avoid them from further splitting. Experimental evaluation on simulated data shows that our algorithm produces parameters which are very close to the original distribution. Clustering on a real data set shows a good correspondence between the clusters and thematic (information) classes chosen by the remote sensing analyst in a supervised classification project. Further analysis and experimentation is needed to understand the performance and utility of this algorithm in earth science data mining applications.

Acknowledgments

We would like to thank our former collaborators Thomas E. Burk, Jamie Smedsmo, Ryan Kirk and Tim Mack at the University of Minnesota for useful comments and inputs into this research. The comments of Eddie Bright, Phil Coleman, and Veeraraghavan Vijayraj, have greatly improved the technical accuracy and readability of this paper. Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285, managed by UT-Battelle, LLC for the U. S. Department of Energy under contract no. DEAC05-00OR22725.

References

1. Feng, Y., Hamerly, G.: Pg-means: learning the number of clusters in data. In: Advances in Neural Information Processing Systems 19, pp. 393–400. MIT Press, Cambridge (2007)
2. Fraley, C., Raftery, A., Wehrens, R.: Incremental model-based clustering for large datasets with small clusters. Journal of Computational and Graphical Statistics 14 (2005)

3. Hamerly, G., Elkan, C.: Learning the k in k-means. In: Neural Information Processing Systems. MIT Press, Cambridge (2003)
4. Jain, A.K., Dubes, R.C.: Algorithms for clustering data. Prentice-Hall, Inc., Upper Saddle River (1988)
5. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Comput. Surv. 31(3), 264–323 (1999)
6. Mclachlan, G.J., Peel, D.: On a resampling approach to choosing the number of components in normal mixture models. In: Proceedings of Interface 96, 28th Symposium on the Interface, pp. 260–266 (1997)
7. Carreira-Perpi, M.A.: Mode-finding for mixtures of gaussian distributions. IEEE Trans. Pattern Anal. Mach. Intell. 22(11), 1318–1323 (2000)
8. Pelleg, D., Moore, A.W.: X-means: Extending k-means with efficient estimation of the number of clusters. In: ICML 2000: Proceedings of the Seventeenth International Conference on Machine Learning, pp. 727–734. Morgan Kaufmann Publishers Inc, San Francisco (2000)
9. Penny, W.: Kullback-liebler divergences of normal, gamma, dirichlet and wishart densities (2001)
10. Rivas, M.: An exposition on tests for multivariate normality (2007)
11. Sankar, A.: Experiments with a gaussian merging-splitting algorithm for hmm training for speech recognition. In: Proceedings of the Broadcast News Transcription and Understanding Workshop, pp. 99–104 (1998)
12. Shapiro, S.S., Wilk, M.B.: An analysis of variance test for normality (complete samples). Biometrika 3(52) (1965)
13. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63(2), 411–423 (2001)

Overcoming Geoinformatic Knowledge Fence: An Exploratory of Intelligent Geospatial Data Preparation within Spatial Analysis

Jian Wang[1], Chun-jiang Zhao[1,*], Fang-qu Niu[2], and Zhi-qiang Wang[2]

[1] National Engineering Research Center for Information Technology in Agriculture (NERCITA), Beijing 100097, China
Zhaocj@nercita.org.cn
[2] Institute of Geography Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing 100060 China
Wangj@nercita.org.cn

Abstract. The booming of earth observation provides decision-makers with more available geospatial data as well as more puzzles about how to understand, evaluate, search, process, and utilize those overwhelming resources. The paper distinguishes a concept termed geoinformatic knowledge fence (GeoKF) to discuss the knowledge-aspect of such puzzles and the approach to overcoming them. Basing on analysis of the gap between common geography sense and geoinformatic professional knowledge, the approach composes analysis space modeling and spatial reasoning to match decision models to the online geospatial data sources they need. Such approach enables automatically and intelligently searching of suitable geospatial data resources and calculating their suitability to given spatial decision and analysis. An experiment with geo-services, geo-ontology and rule-based reasoning (Jess) is developed to illustrate the feasibility of the approach in scenario of data preparation within decisions of bird flu control.

Keywords: spatial decision support, geo-ontology, spatial reasoning, geo-services, geography information metadata.

1 Introduction

With booming of earth observation and other earth-related activities, there is a dramatically increase of types and volume of available geospatial data in form of online data set and services. Companying with such overwhelming data, users, esp. those without enough geoinformatic knowledge (termed here as n-geo users), find it more difficult to understand and utilize those dazzling resources [1-3]. Most difficulties can finally find the root to insufficiency of geoinformatic knowledge (such as those relating to temporal-spatio scale or precision) and so are defined here as geoinformatic knowledge fence (GeoKF). With continually increasing of available geospatial data, GeoKF will

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 385–394, 2009.

become an important obstacle to earth observation application, geospatial information services and other geocomputing applications.

The experiment is an exploratory to overcome GeoKF by combining geoinformatic knowledge with computing reasoning in scenario of geospatial data preparation within spatial decision making. Five sections are laid out in the paper. Section 1 introduces and defines GeoKF, then section 2 explains the philosophy and conceptual model of the approach to overcoming GeoKF. In section 3 the design of experiment system is detailed following the conceptual model, the experiment result is analyzed in the same section. Finally there are a discussion (section 4) and a conclusion (section 5) at the end of the paper.

2 Philosophy and Conceptual Model of GeoKF Overcoming

2.1 Philosophy of GeoKF Overcoming

A questionnaire on 30 n-geo experts (mainly in agriculture engineering, information system and management) of NERCITA shows two sorts of geographic knowledge during their spatial analysis. One can be termed as common geographic sense (CGS) mainly by which people model their analysis space through assigning i.e. spatio-temporal domain, scale and precision, etc.; the other can be termed as geoinformatic professional knowledge (GPK) by which geoinformatic professionals can understand various geospatial data and utilize them in a more effective and suitable way, for example processing some ostensibly unsuitable data by generalization or scale transformation to meet spatial analysis. Most n-geo experts lack the later and so are prevented from efficiently geospatial data preparation within their spatial analysis.

The research philosophy comes from the knowledge-oriented analysis of GeoKF and spatial decision process (Fig.1). GeoKF roots from knowledge heterogeneity between n-geo users and geoinformatic professionals, and can be represented as a fence between CGS and GPK. CGS, together with domain knowledge, is generally employed by domain professionals (most are n-geo users) to construct space model of analysis (SMoA) within which the target objects or processes are modeled and analyzed [4]. Such models also determine the suitable geospatial data for analysis, namely, record the data requirements (both data sources and their properties) in form of CGS. On the other side, various geospatial data is traditionally described and represented by their GPK-based metadata in terms of i.e. FGDC. So GeoKF-overcoming practically means the mapping between SMoA and GPK-based metadata, which generally perform by cooperation of geoinformatic professionals and n-geo SMoA makers.

The paper aims to find an intelligent and automatic approach to replacing the cooperation by spatial reasoning. Specifically, the paper aims to verify two hypothesizes below.

- Hyp. #1. CGS can be regarded as dialect subsets of GPK, which means the data requirements of CGS-based SMoA can be 'translated' into GPK-based description of geospatial data. The translation gives the possibility of spatial reasoning to match SMoA and the data it need.
- Hyp. #2. Current GPK-based metadata can be modified or extended to support spatial reasoning relating to down or up-scaling, precision transform, etc.

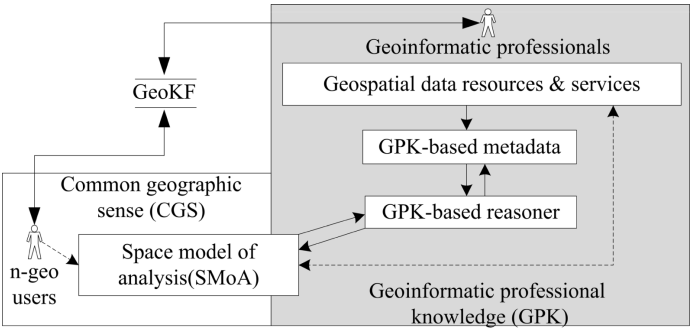


Fig. 1. GeoKF-overcoming philosophy

2.2 Conceptual Model

Space Model of Analysis. From the perspective of n-geo users, each spatial analysis task has its space models, which contain all analyzed geography features within certain spatial and temporal domain. The model can be expressed by a triple of (*features, space, time*). *Features* can be further expressed as (*focal features, background features*). *Focal features* are those analyzed such as roads in transport analysis; *background features* are those that function as context of *focal features* such as land use in transport analysis. Each dimension (*space* or *time*) is identified by domain, scale and precision. Although some geoinformatic professionals suggest that scale covers precision [5-7], most n-geo users are apt to confuse them and so the research employs both concepts contemporarily. Scale here means the size of analyzed or sampled unit. Precision here refers to minimum certain measure value of analysis. By this way, *space* dimension is the triple of (*spatial domain*, *spatial scale*, *spatial precision*), while *time* dimension is another triple of (*temporal domain*, *temporal scale*, *temporal precision*).

Table 1. Some main fundamental ontology

Fundamental ontology	Description
Geospatial theme-feature ontology	Extended ISO 19115[9], one example of extension are poultry feature of agriculture theme.
Geospatial scale and precision ontology	Describing the knowledge of geospatial scale and precision, including, metric scale (i.e. mapping scale, traditional airphoto scale, etc.), named scale (Global scale, continent scale, region scale etc.), and precision.
Geospatial domain ontology	Describing the knowledge of geospatial domain, including named domain (government precinct, geology zoning, geography zoning, etc.)
Time scale and precision ontology	Describing the knowledge of time scale and precision, including metric scale (second, day, year, etc.), named scale (Macro scale, geography scale, etc.), and precision (second, minute, …, year, etc.)
Temporal domain ontology	Describing temporal domain, including named domain (geology, geography, etc.)

Knowledge Organization and Reasoning. All knowledge (CGS & GPK) should be organized in computable formats such as ontology, knowledge base with facts and rules[8], etc. The organization should consider both computing (i.e. spatial reasoning) and human operating. Some main fundamental knowledge is listed in Tab. 1 in form of ontology.

Spatial Reasoning. The spatial reasoning (Fig. 2) combines human intelligence and computing reasoning by an iteration cycle consisting of 1) rendering data requirements of SMoA to GPK-based reasoner, 2) reasoning and suitability calculating basing on data requirements, GPK, and metadata, and 3) returning result to n-geo users, who can re-render or accept result to end current cycle. Each cycle consists of three sub-reasoning tasks (theme-ontology evaluation, space evaluation, and time evaluation). Each task reasons on a section of SMoA requirement, corresponding knowledge base (represented as ontology in Fig.) and metadata of evaluated geospatial data source. The performance of each task will return the section suitability of current geospatial metadata. GPK-based resoner controls the running of sub-task and calculates the whole suitability of evaluated geospatial data sources by scanning metadata list of them. Formula (1) is used to calculate the suitability index (*SI*), while some GPK-based reasoning rules are listed in Tab.2.

$$SI=\sum((suitability\text{-}item)_i\times W_i)(i=1,\ldots m) \quad (1)$$

SI: suitability index, the final suitability value of evaluated geospatial data.

$(Suitability\text{-}item)_i$: the result of evaluating of certain *(i)* items of SMoA.

W_i: the weight of each items of SMoA. The valued is appointed by decision makers to assign the importance of items. The total of all weight is 100%.

m: the number of evaluated items of SMoA.

Table 2. Main rules for suitability reasoning and calculation

Items of SMoA	Suitability-item	Description
Feature	*M/N×100%*	*N*: the number of expected features in SMoA *M*: the number of existence features of evaluated geospatial data
Space-Domain	*(SMoA-D/Data-D)×100% (0-100%)*	*SMoA-D*: the domain of spatial analysis *Data-D*: the domain of evaluated geospatial data
Space-Scale & Time-scale	*0%, 100%*	100%: the scale of evaluated geospatial data is equal directly or by upscaling. 0%: other occasions
Space-Precision & Time-Precision	*0%, 100%*	100%: the precision of evaluated geospatial data directly or more precise to SMoA. 0%: other occasions
Time-domain	*0%, 100%*	100%: the time period of evaluated data content can cover time of SMoA completely 0%: other occasions

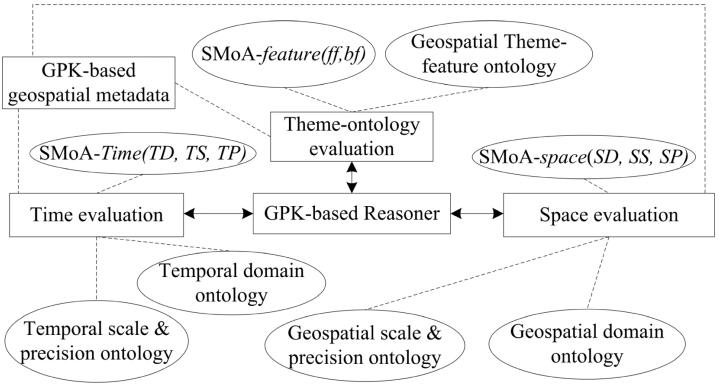


Fig. 2. Spatial reasoning

Conceptual Model. The model have 5 key components including geospatial data source agent, n-geo users and geoinformatic professionals, knowledge base (facts and rules), GPK-based reasoner, and human-computer interface (Fig.3). Geospatial data sources agent functions as a list of enrolled GPK-based metadata (Tab.3) of online

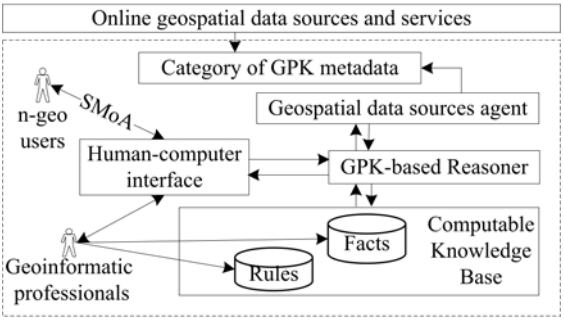


Fig. 3. Conceptual Model

Table 3. GPK-based FGDC metadata (limited to extended or modified items only)

FGDC metadata item	Extended or modified metadata Items	Description of extension or changes
Entity & Attributes and Keyword-Theme	Theme-features	Bounding Entities to Theme-feature ontology
Identification-spatial domain	Spatial domain	Bounding to geospatial domain ontology
Null	Spatial Scale	Bounding to geospatial scale & precision ontology
Null	Temporal scale	Bounding to temporal scale & precision ontology

geospatial data sources. CGS and GPK are organized in computable knowledge base for GPKreasoning. Human-computer interface is a GUI web page that lets decision makers (mainly n-geo users) render (and re-render) their SMoA as well as check the suitability of reasoned results.

3 Experiment Design and Analysis

An experiment, with a software system as its core, is designed and implemented to check the feasibility of philosophy. The comparison of reasoned results with evaluation of both n-geo users and geoinformatics professionals is employed as the basic method for the verification.

3.1 Experiment Design

The experiment is a simplified spatial decision support of bird flu control in an agri-intensive county of Beijing area. Three independent and geographically distributed government agencies, which hold different geospatial and agricultural data sources, will collaborate with their data sources. The spatial analysis aims to zone epidemic impacted area and set block stations to control epidemic diffusion. Some main items of SMoA of the decision are listed in Tab.4. The available online geospatial data can be seen in Tab.5.

Table 4. SMoA items of bird flu control decision

SMoA items	Value	Description
Feature-focal	Poultry – livestock - agriculture	From geospatial theme-feature ontology
Feature-background	Roads-Transportation	From geospatial theme-feature ontology
Feature-background	Resident area - land use - plan	From geospatial theme-feature ontology
Feature-background	Government precinct – named domain	From Geospatial domain ontology
Space-domain	Daxing District-Beijing city-China-Government precinct-Named domain	From geospatial domain ontology
Space-Scale	1:10000, Mapping scale-metric scale	From geospatial scale & precision ontology
Space-Precision	100M – precision	From geospatial scale & precision ontology
Time-domain	after. 2004	Null
Time-scale	Year-metric scale	From Temporal scale & precision ontology
Time-precision	Null	Null

Software System Design. As an experiment prototype, the system adopts Jess and Rete algorithm [8] because that Jess is a classic and fundamental technology for knowledge organization and reasoning. Ontology, and other knowledge technologies, is only used to organize knowledge to apart the proposed philosophy from implementation technologies as possible as it can. The logic structure of software system is listed in Fig 4.

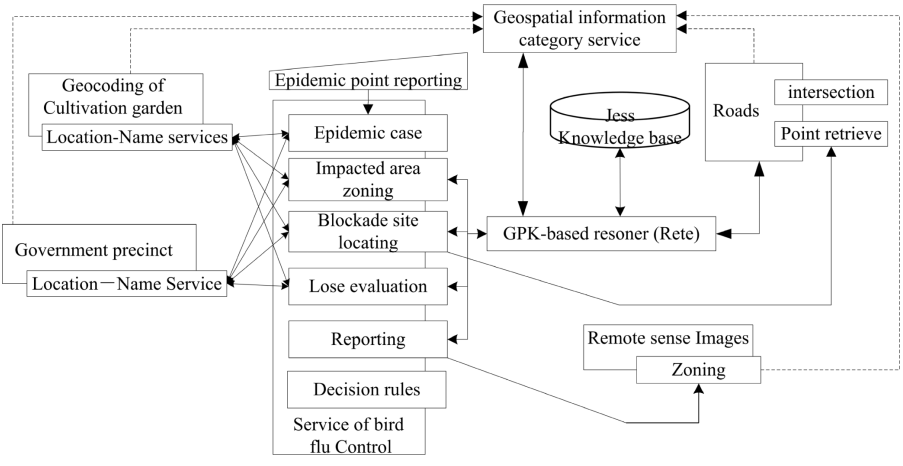


Fig. 4. The system is an implementation (with 7 web services and 13 endpoints) of conceptual model (Fig.3). The service *geocoding of cultivation garden*, *government precinc*t, *roads*, and *remote sense images* are all online geospatial data services and enrolled into *geospatial information category service*. The *service of bird flu control* is an application service (with a GUI page) that performs the spatial analysis trigged by a geo-referenced *epidemic point reporting*. The endpoint of *epidemic case* will initialize the spatial analysis of bird flu control as well as construct general SMoA basing on epidemic report. The endpoint of *impacted area zoning*, *blockade site locating*, *lose evaluation* and *reporting* will undertake the spatial analysis within which their detailed SMoA will be constructed basing on general SMoA and then its requirements will be transported o *GPK-based reasoner* for obtaining the suitable geospatial data sources. The reasoner will return reasoned result to corresponding endpoint for refining if needed.

3.2 Result and Analysis

Tab. 5 shows the comparison of reasoned results with human evaluation. Such comparison displays a generally consistence between reasoned results and expert evaluations, which can preliminarily give feasibility for further research. While as a philosophy-proof experiment, there still exist many defects such as small sample of online geospatial data, relatively simple reasoning rules (esp. on scale down- and up-scaling and precision, etc.) and facts, etc. Additionally, a strong suggestion from experts for further research is to add quality of data services (QoS) into SMoA and GPK-reasoning to get dynamic optimal data set.

Table 5. Comparison of human evaluation with reasoned result

Available geospatial data	Human evaluation of suitability	Reasoned suitability index(%)								
		total	Feature%		Space			Time		
			FF 30%	BF 5%	SD 30%	SS 10%	SP 5%	TD 10%	TS 5%	TP 5%
Cultivation garden (2004)	High	95	100	0	100	100	100	100	100	100
Fundamental geography information (2000: 1:10M)	Mid	85	50	100	100	100	100	100	100	100
Beijing fundamental geography informa-tion(1998 1:10000)	High	75	50	100	100	100	100	0	100	100
Beijing Quickbird image (2008-12: R=0.61M)	High	100	100	100	100	100	100	100	100	100
Beijing DEM (1:10000, 2001)	Low	0	0	0	100	Null	Null	Null	Null	Null
Beijing MODIS (2008-12, 30M)	Mid	70	50	100	100	0	0	100	100	100

4 Discussion

According to the exploratory research, GeoKF-overcoming is a synthetic issue relating to some traditional geocomputing problems including intelligent geospatial data generalization [10-11], scale and precision transformation [12], etc. In other words, GeoKF research will depend on and contribute to those relating studies. Such relationships among them also discover some features of further researches.

- The combination of artificial and human intelligence. Because of the immature of scale transform, automatically geospatial data generalization and other relative issues, the overcoming of GeoKF is still a combination of human and computing intelligence, namely, increasing the level of automation by adding more knowledge and reasoning ability is a key research direction.
- The dependence of GeoKF-overcoming practice on fundamental GPK research and practices. Although more and more discussions of geo-ontology or other geo-knowledge relative researches appear recently, the practice works, such as construction

of geo-ontology (including geospatial theme-feature ontology, geospatial scale & precision ontology, etc.), have not yet functioned. Furthermore, the workings relating to geo-knowledge components, such as adding more semantic to metadata to support computing reasoning, are still in their way to practice. It is clear that the power of proposed approach of the paper will grow with increase of those workings.

5 Conclusion

With booming of available online geospatial data resources, the bottleneck of net-centric geocomputing has moved gradually from network and computing capability to efficiently utilization of data resources by more users, esp. n-geo users. As a consequence, geoinformatic knowledge and intelligent geo-computing (including geo-ontology, spatial reasoning, intelligent generalization of geospatial data, etc.) has been attracting increasing attentions when facing overwhelming online resources.

The research suggests that GeoKF is an emerging issue within intelligent geo-computing. It shares one aim of geo-cognition research, namely, how to provide n-geo users with correct understanding and utilization of dazzling online geospatial data sources, as well as to help geoinformatic professionals improve their efficiency and capability. Featuring with knowledge-oriented analysis of geospatial data process and spatial analysis, the conceptual model with SMoA, GPK-based reasoning and extended GPK metadata, and the combination of human and computing intelligence, the proposed approach shows its feasibility for further research of GeoKF.

Acknowledgments. The paper is supported by Beijing sci-tech Nova plan (2006B27) and national key technology R&D program (2006BAJ05A09, 2006BAD10A05).

References

1. Rahimi, S., Cobb, M., Ali, D., Paprzycki, M., Petry, F.E.: A knowledge-based multi-agent system for geospatial data conflation. Journal of Geographic Information and Decision Analysis, 1480–8943 (2002)
2. Wei, Y., Yue, P., Dadi, U., Min, M., Hu, C., Di, L.: Effective Acquisition of Geospatial Data Products in a Collaborative Grid Environment. In: IEEE International Conference on Services Computing (SCC 2006), pp. 455–462. IEEE Press, New York (2006)
3. McMaster, R.B., Lynn, E.: A Research Agenda for Geographic Information Science. CRC Press, Boca Raton (2004)
4. Longley, P., Batty, M.: Advanced Spatial Analysis. ESRI Press, Redlands (2003)
5. zhilin, L.: A theoretic discussion on the scale issues in geospatial data handling. J. Geomatics world, 1–5
6. Sun, q.-x., Li, m.-t., Lu, j.-x., Guo, d., Fang, t.: Scale issue and its research progress of geospatial data. J. Geography and geo-information science 23, 5–56 (2007)
7. Meng, b., Wang, j.-f.: A review on the methodology of scaling with Geo-data. J. ACTA geographic SINICA 60(2), 277–288 (2005)
8. Friedman-Hill, E.: Jess in Action: Java Rule-Based Systems. Manning Publications Co. (2003)
9. ISO/TC211 19115: Geographic information – Metadata (2003)

10. Lehto, L., Sarjakoski, L.T.: Real-time generalization of XML-encoded spatial data for the Web and mobile devices. J. Geographical information science 19(8,9), 957–973 (2005)
11. Jones, C.B., Abdelmoty, A.I., Lonergan, M.E., van der Poorten, P., Zhou, S.: Multi-scale spatial database design for online generalisation. In: Proceedings of the Spatial Data Handling Symposium, Beijing, pp. 7b.34–7b.44 (2000)
12. Li, S.-l., Cai, y.-l.: Some scaling issues of geography. J. Geographical research 1, 11–18 (2005)

Spatial Relations Analysis by Using Fuzzy Operators

Nadeem Salamat and El-hadi Zahzah

Université de La Rochelle
Laboratoire de Mathématiques, Images et Applications
Avenue M Crépeau La Rochelle 17042, France
{nsalam01,ezahzah}@univ-lr.fr
nadeemsalamat@hotmail.com

Abstract. Spatial relations play important role in computer vision, scene analysis, geographic information systems (GIS) and content based image retrieval. Analyzing spatial relations by Force histogram was introduced by *Miyajima et al* [1] and largely developed by Matsakis [2] who used a quantitative representation of relative position between 2D objects. Fuzzy Allen relations are used to define the fuzzy topological relations between different objects and to detect object positions in images. Concept for combined extraction of topological and directional relations by using histogram was developed by J.Malki and E.Zahzah [3], and further improved by Matsakis [4]. This algorithm has high computational and temporal complexity due to its limitations of object approximations. In this paper fuzzy aggregation operators are used for information integration along with polygonal approximation of objects. This approach gives anew, with low temporal and computational complexity of algorithm for the extraction of topological and directional relations.

Keywords: Spatial Relations, Force Histogram, Polygonal Approximation, Temporal Complexity, Fuzzy aggregation operators.

1 Introduction

Space relations has a remarkable importance in computer vision and image analysis as in content based image retrieval, similarity based image retrieval, identify forms, manage data bases, support spatial data in artificial intelligence (AI), cognitive science, perceptual psychology, geography particularly geo-information system (GIS), indexation and comparing objects scene and model are major applications of space relations. Different approaches for finding spatial and topological relations have been developed according to the need for applications and object representations. Qualitative methods for directional and topological relations includes Max J. Egenhofer's method of four and 9 intersections [5,6]. These methods are considered most important in GIS community. Directional relations are defined on relative frame of reference and absolute frame of reference. In relative frame of reference position of a simple object is made with respect to an

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 395–404, 2009.

oriented line or an ordered set forming a vector to some intrinsic properties of reference object. Methods like angle histogram introduced by K.Miyajima and A.Ralescu [1] and statistical method developed by MinDeng.Zalimli [7], $R - histogram$ [8] depends upon relative reference frame. Matsakis [2] introduced 1D representation of 2d objects by the union of longitudinal sections which is the extension of angle histogram. The derivation of cobined topological and directional relations by using force histogram [4] was first introduced by J.Malki and E.Zahzah [3], then Matsakis raised some problems regarding fuzziness of some relations like *meet* and *meet_by* and some others which exist at segmentation level. In case of longitudinal section fuzzification process introduced by Matsakis [4] restricts the object approximation which increases the temporal and computational complexity of algorithm.

Approximating the object by its boundary, length of longitudinal sections can be computed as distance between the intersecting points of oriented line and object boundary. The degree of fuzzy membership function depends upon three values x, y and z. In tuple (x, y, z), the pair (x, z) are the lengths of longitudinal sections and y is the difference between maximum value of intersecting points of object B and minimum value of object A, i.e. $y \in R$ or $y \in Z$. By this approach of object approximation, temporal complexity decreases from $n\sqrt{n}$ to $N \log(N)$ where n is number of pixels of objects under consideration and N is the number of vertex of object polygons. Temporal complexity for the said algorithm is not given but in general temporal complexity of force histograms is discussed in [9] for different object types. We assume same temporal complexity for objects because segmentation level problems raised by Matsakis forced the object as raster data and in addition to this algorithm for fuzzification of longitudinal sections increases temporal and computational complexity. These problems no more exist if we consider objects by their boundary, then need for Matsakis's algorithm remain for objects having disconnected boundaries. Each segment is separated by a certain distance. This internal distance has a significant impact on directional and topological relations. Fuzzy disjunctive operators are used. These fuzzy operators have been developed to summarize information distributed in different sets by grades of fuzzy membership values. This paper is structured as follows. First of all we describe Allen relations in space. In section 2 we describe different fuzzy Allen relations defined by Matsakis and changes in mathematical formulation of fuzzy histogram of Allen relations due to object approximation. In section 3 we discuss different fuzzy operators, section 4 describes experimental results. In section 5 temporal complexity is calculated and section 6 concludes the paper.

2 Allen Relations in Space

Allen [10], introduced the well known 13 mutually exclusive exhaustive interval relations based on temporal interval algebra. These relations are arranged as $A = \{<, m, o, s, f, d, eq, d_i, f_i, s_i, o_i, m_i, >\}$. where $\{<, m, o, s, f, d, \}$ $(\{d_i, f_i, s_i, o_i, m_i, >\})$ are the relation $before$, $meet$, $overlap$, $start$, $finish$, $during$ (resp the inverse relations of the cited ones). The relation eq is the equality spatial relation.

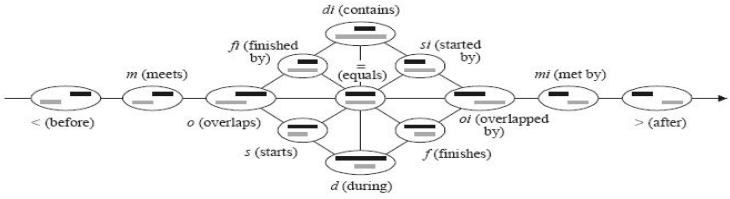


Fig. 1. Black segment represents the reference object and gray segment represents argument object. figure extracted from [2].

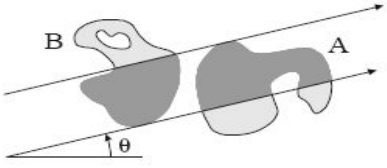


Fig. 2. Histogram of Allen in direction θ (dark gray area represents histogram of fuzzy Allen relations, figure extracted from [2])

All the Allen relations in space are conceptually illustrated in figure 2. These relations have a rich support for the topological and directional relations.

3 Fuzzy Histogram of Allen Relations

In real applications, small errors in crisp values can change the entire result when gradual changes of topological relations occur over time. To cope these problems fuzzification was introduced, it comprises the process of transforming crisp values into grades of membership for linguistic terms of fuzzy sets. Fuzzification process of Allen relations do not depend upon particular choice of fuzzy membership function, trapezoidal membership function is used due to flexibility in shape change. Let $r(I, J)$ is Allen relation between segments I and J where $I \in A$(argument object) and $J \in B$ (reference object), r' is the distance between $r(I, J)$ and its conceptional neighborhood. We consider a fuzzy membership function $\mu : r' \longrightarrow [0, 1]$. The different fuzzy Allen relations defined by Matsakis [4] are

$$f_b(I,J) = \mu_{(-\infty,-\infty,-b-3a/2,-b-a)}(y)$$

$$f_m(I,J) = \mu_{(-b-3a/2,-b-a,-b-a,-b-a/2)}(y)$$

$$f_f(I,J) = min(\mu_{(-(b+a)/2,-a,-a,+\infty)}(y), \mu_{(-3a/2,-a,-a,-a/2)}(y), \mu_{(-\infty,-\infty,z/2,z)}(x))$$

$$f_{fi}(I,J) = min(\mu_{-b-a/2,-b,-b,-b+a/2}(y), \mu_{(-\infty,-\infty,-(b+a)/2)}(y), \mu_{(z,2z,+\infty,+\infty)}(x))$$

$$f_d(I,J) = min(\mu_{(-b,-b+a/2,-3a/2,-a)}(y), \mu_{(-\infty,-\infty,z/2,z)}(x))$$

$$f_{di}(I,J) = min(\mu_{(-b,-b+a/2,-3a/2,-a)}(y), \mu_{(z,2z,+\infty,+\infty)}(x))$$

where $a = min(x, z)$, $b = max(x, z)$, x is the length of longitudinal section of argument object A, and z is the length of longitudinal section of reference object B. Most of relations are defined by one membership function and some of them by the minimum value of more than one membership functions like $d(during)$, $d_i(during\_by)$, f $(finish)$, f_i $(finished\_by)$. In fuzzy set theory, sum of all the relations is one, this gives the definition for fuzzy relation *equal*. Histogram of fuzzy Allen relations stated by Matsakis [4] is "Histogram of fuzzy Allen relations represents the total area of subregions of A and B that are facing each other in given direction θ".

In this new approach, fuzzy Allen relations are computed for each segment. Fuzzy Allen relation for each segment is a fuzzy set and fuzzy aggregation operators are used to combine different values of fuzzy grades. This results the change in above definition of fuzzy histogram of Allen relations. Mathematically this becomes

$$\int_{-\infty}^{+\infty} (\sum_{r \epsilon A} F_r(q, A_q(v), B_q(v))dv = (x + z) \sum_{k=1}^{n} r(I_k, J_k) \tag{1}$$

where z is the area of reference object and x is area of augmented object in direction θ, n is total number of segments treated and $r(I_k, J_k)$ is an Allen relation for segmentsI_k, J_k.

4 Fuzzy Operators and Treatment of Longitudinal Sections

During the decomposition process of an object into segments, there can be multiple segments depending on object shape and boundary which is called longitudinal section. Different segments of a longitudinal section are at a certain distance and these distances might effect end results. After polygon object approximation, we need for the fuzzification algorithm when object has disconnected boundary. In this case there exist number of 1D segments of concave object or object having disconnected boundary. Each segment and its distance from other segment has its own impact on fuzzy Allen relations of whole object. To cope with this, fuzzy operators are used. In literature of fuzzy set theory there exist variety of operators such as fuzzy $T - norms$, $T - conorms$ and so on, which can be used for fuzzy integration of available information. Some mostly used operators for data integration in [11]are:

$\mu_{(OR)}(u) = max(\mu_{(A)}(u), \mu_{(B)}(u))$; $\mu_{(AND)}(u) = min(\mu_{(A)}(u), \mu_{(B)}(u))$;
$\mu_{(SUM)}(u) = 1 - \Pi_{i=1}^{2}(\mu_{(i)}(u))$,

When fuzzy operator OR (respectively AND) is used, only one fuzzy value contributes for the resultant value which is *maximum* (respectively *minimum*). For other operators both values contribute. In this case each Allen relation has a fuzzy grade objective is to accumulate the best available information. In case of longitudinal section, there exist number of segments and each segment has

a fuzzy Allen relation with segment of other object. Suppose that longitudinal section of object B has two segments such that $z = z_1 + z_2$ where z_1 is the length of first segment and z_2 is the length of second segment and z is length of longitudinal section. Let $\mu_1(y_1)$ defines the value of fuzzy Allen relations with the first segment and $\mu_2(y_2)$ represents value of fuzzy Allen relations with the second segment where y_1 and y_2 are the distances between object A and two segments of B. Now Fuzzy OR operator is used to get consequent information obtained from two sets of fuzzy Allen relations.

5 Experiments and Interpretation

For the experiment purpose 360 directions are considered (angle increment is 1 degree) and lines are drawn by 2d Bresenham digital line algorithm. Instead of considering all the v values, only those lines are considered which passes through vertex of polygon. segments are computed and fuzzy Allen relations are computed for each segment, if there exit longitudinal section then fuzzy aggregation operator is applied to obtain the resultant fuzzy Allen relation of whole object. Each relation is associated with the gray scale value like $before$ with black and white represents $after$. Same notations as Matsakis are used except changing the boundary color of a relation for better visualization of relations. Opposite relations have the same boundary color such as $m(meet)$ and $(meet\_by)$ relations have the yellow boundary. Object A has the light gray color while object B is represented by dark gray. The thirteen histograms that represent directional and topological relations are represented in the layers and each vertical layer represents total area of objects in that direction. Here histograms are not normalized. All relations are symmetric in nature except $d(during)$ and $di(during\_by)$.

$f_b^{AB}(\theta) = f_a^{AB}(\theta + \pi)$, $f_{mi}^{AB}(\theta) = f_m^{AB}(\theta + \pi)$, $f_{oi}^{AB}(\theta) = f_o^{AB}(\theta + \pi)$, $f_{si}^{AB}(\theta) = f_{fi}^{AB}(\theta + \pi)$, $f_f^{AB}(\theta) = f_s^{AB}(\theta + \pi)$ and for $d(during)$, $di(during\_by)$ it will be $f_d^{AB}(\theta) = f_{di}^{BA}(\theta)$.

In fig.3(a) explains the representation of fuzzy Allen histograms.In fig.3(b) Shows the representation of histograms and explains that each relation is represented by a layer and each layer have a different gray level color associated with a relation, boundary color is not represented here. Same colors association with a relation is used only boundary colors are changed. In fig.3(c) represents object position where A is light gray object and object B is represented by dark gray color. Fig.3(c) represents histogram associated with objects pair, where y axis represents total area of objects having different relations and directions are represented along x axis. At a certain value f represents area under the $finish$ relation and d represents area having $during$ relation and total area is sum of both areas. Different set of examples are considered, in first case both the objects are convex and second case argument object A is convex and reference

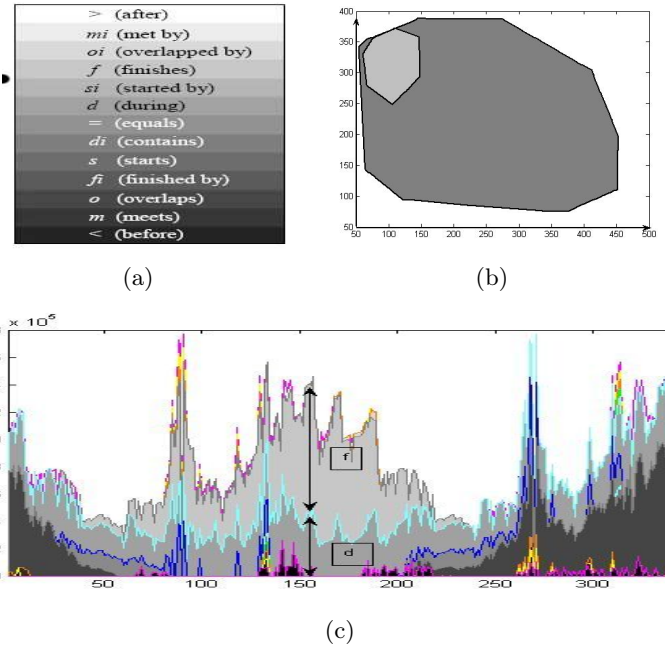


Fig. 3. (a)Explanation of gray level value associated with a relation(source: [4]) (b)Object pair representation. (c) Corresponding histogram.

object B is concave. In this experiment Fig.4(a) Pair of objects under consideration are at enough distance. Fig.4(b) represents the corresponding fuzzy histogram of Allen relations, at this stage only $after$ and $before$ fuzzy relations exist. Fig 4(e) object A moves to words center of object B and it overlaps B. Fig.4(d) represents its histogram at this almost all the relations exist. Fig.4(e) position of object A is at center of object B. Fig.4(f) represents its histogram $during$ relation exist, There exist $after$ and $before$ relation near the diagonal direction. Which is due to zigzag of lines in digital space. In this set of examples objects are taken at different distances to show that the relations are sensitive to distance between them and their sensitivity also depends upon relative size.

Now for second set of examples. In this example rectangular objects are considered. Object A firstly for away from the U shaped object B. Fuzzy Allen relations are calculated separately for each segment then fuzzy operator is used. Main objective of this example is to show that each segment of longitudinal section has its own impact on fuzzy Allen relation and each segment may have same, opposite neither opposite nor same Allen relations as in case of fig.5(a)to fig.5(c). In Fig.5(a) object A is at a certain distance to object B. Fig.5(b) only $after$ and $before$ relation exists because both parts of object B has the same relation. Finally in fig.5(c) when object A is between two parts of object B,

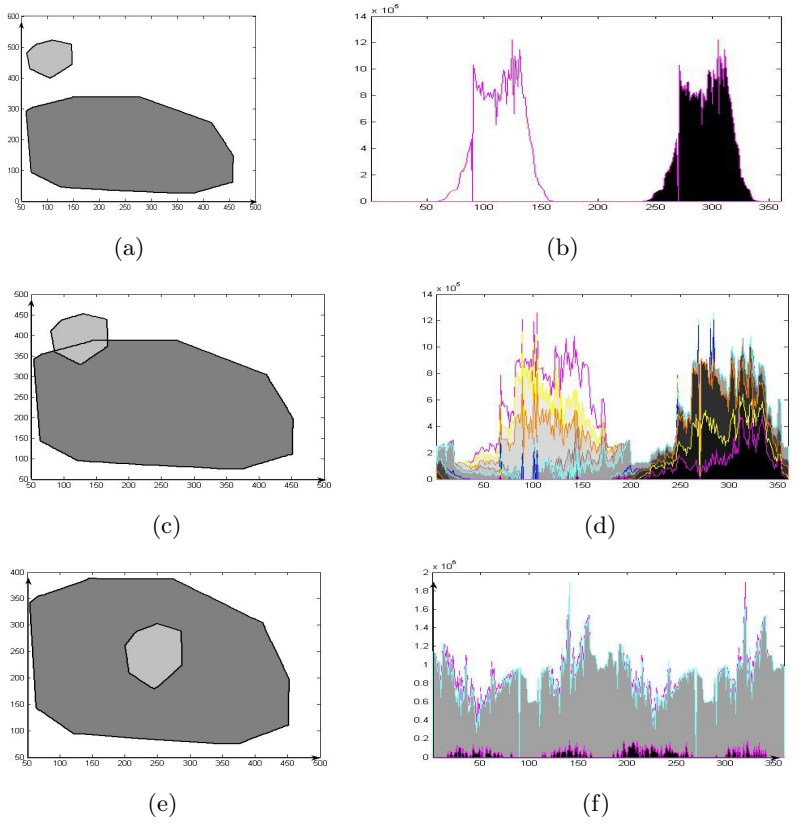


Fig. 4. Pairs of convex objects (where along x-axis angle is zero degree) and their corresponding fuzzy histogram of Allen relations

both segments have opposite relations $before$ and $after$ meanwhile there exist relation *during* which is due to zigzag phenomena of digital space and line algorithm.

6 Temporal Complexity

Finding the exact temporal complexity is a tough job, major aim of this study is to find time length required by the algorithm. To expresse the computational time as a function of N a language is required which grows on the order of N. Five symbols for comparing rates are used such as o, O, θ, Ω and $\sim$. In fact asymptomatically equality is an formalism of idea to find the conditions that two functions have same growth rate.i.e. $limit_{n\to\infty}(\frac{a_n}{b_n}) = 1$ and $a_n = O(b_n)$ if $|\frac{a_n}{b_n}|$is bounded. Asymptotic analysis algorithm is used. For this purpose a function which represents upper bound of function used to represent the algorithm complexity is found. In this case time constraint depends upon length of line,

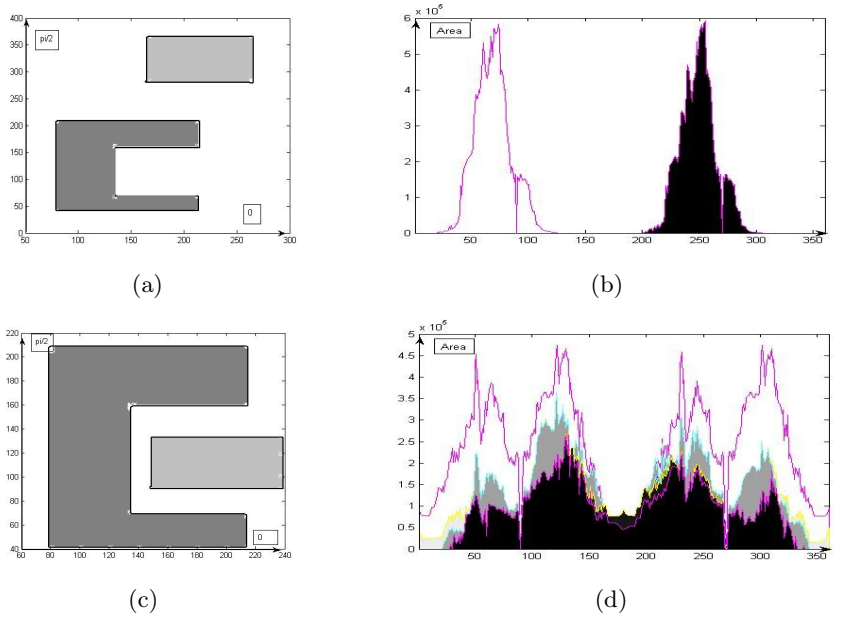


Fig. 5. Convex and concave pair of objects (where along x-axis angle is zero degree) and their corresponding fuzzy histogram of Allen relations

contour length and number of polygon vertexes. Time (round in seconds) for all 360 directions is used when line length is fixed to 1000, 1200, 1400 pixels. Following tables represent different computations where L is length of line and N represents number of polygon vertexes. There is a symmetry between the different values of cost function (Time function) in table 1 and table 2 and the number of polygons vertexes. By observing the graphical representation of data, (graph fig.6(a)) in table 1 and graph fig.6(b) of data in table 2) each time graph for a fixed length of line and given objects sizes (length of contours). It seems that graph is displaced by a constant value of T and its growth rate is less than $nlog(n)$ hence function $f(n) = nlog(n))$ representation the upper bound of our graphs. (Graphes given in figure.6) So histogram of fuzzy Allen relations are of order $O(Nlog(N))$

Table 1. Contour of 1300 pixels

$N \diagdown L$	1000	1200	1400
24	63	67.14	74
25	66.5	73	78
26	68	74.3	80
27	72	79	84.34
28	73.20	82	86.20

Table 2. Contour 3300 of pixels

$N \diagdown L$	1000	1200	1400
25	97.5	103	113.5
26	102	107	115.5
27	109	115.5	123
28	113	119	126
29	120	125	129.4
30	124	129	135

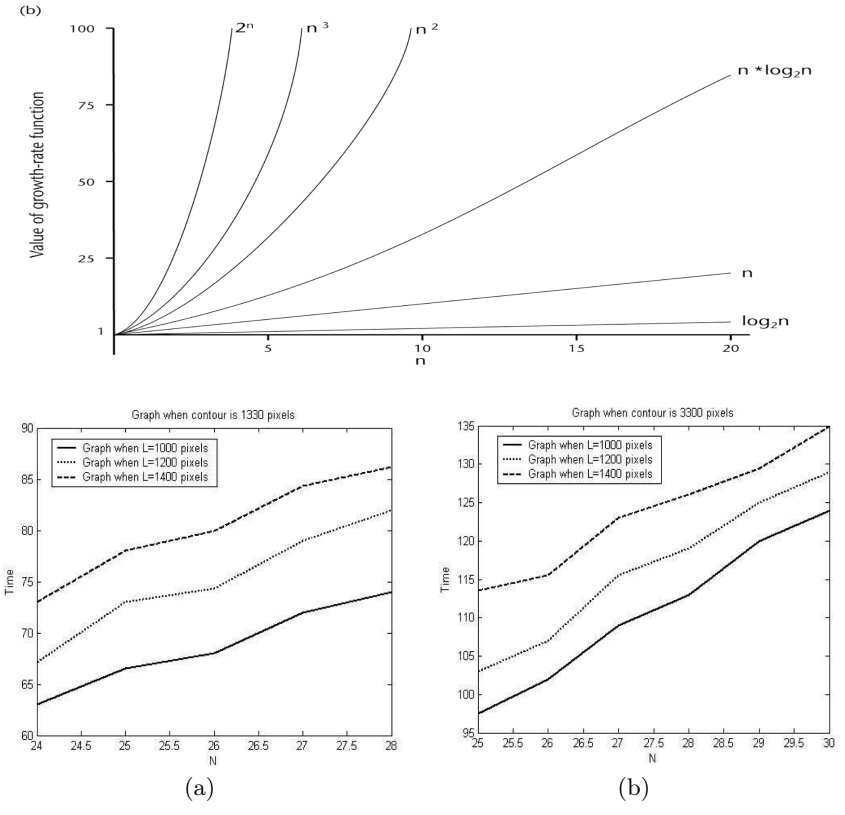


Fig. 6. (a)Graph of some known functions, (b)Graphs of data given in table No.1(c)Graphs of data given in table No.2 where time is rounded off in seconds

7 Conclusion

It is shown that histogram of fuzzy Allen relations associated with pair of objects carry a lot of information. To deal with concave objects or objects having disconnected boundaries, fuzzy operators are used. Use of these operators is simple so polygonal approximation of objects and application of fuzzy aggregation operator simplifies the algorithm given by Matsakis [4]. This approach decrease its temporal and computational complexity due to avoiding the fuzzification process developed by Matsakis. Certainly this approach of using fuzzy operator will open new fields of applications for fuzzy aggregation operators. Here we calculated all the directions for experimental purpose, in practice we performed only limited number of directions according to the requirement of application. Allen relations are used for describing the relative object position in image understanding.

References

1. Miyajima, K., Ralescu, A.: Spatial Organization in 2D Images, Fuzzy Systems. In: IEEE World Congress on Computational Intelligence, vol. 1, pp. 100–105 (1994)
2. Matsakis, P., Laurent Wendling, J.D.: Représentation de La Position Relative d' Objets 2D au Moyen D'Un Histogramme de Forces. Traitement du Signal 15, 25–38 (1998)
3. Malki, J., Zahzah, E., Nikitenko, D.: Indexation et Recherche d' Image Fondées Sur Les Relations Spatiales Entre Objets. Traitement du Signal 19(4), 235–250 (2002)
4. Matsakis, P.: Combined Extraction of Directional and Topological Relationship Information from 2D Concave Objects, in fuzzy modeling with spatial informations for geographic problems, New York, pp. 15–40 (2005)
5. Egenhofer, M.J., Franzosa, R.D.: Point Set Topological Relations. International Journal of Geographical Information Systems 5(2), 161–174 (1991)
6. Egenhofer, M.J., Sharma, J., Mark, D.M.: A Critical Comparison of The 4-Intersection and 9-Intersection Models for Spatial Relations: Formal Analysis. Auto-Carto 11, 1–12 (1993)
7. Li, M.D.: A Statistical Model for Directional Relations Between Spatial Objects. GeoInformatica 12(2), 193–217 (2008)
8. Wang, Y., Makedon, F.: R-histogram:quantitative representation of spatial relations for similarity-based image retrieval. In: MULTIMEDIA 2003, pp. 323–326. ACM, New York (2003)
9. Pascal Matsakis, D.N.: Applying Soft Computing in Defining Spatial Relations, Understanding the Spatial Organization of Image Regions by Means of Force Histograms A Guided Tour, pp. 99–122. Springer, New York (2002)
10. Allen, J.F.: Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26(11), 832–843 (1983)
11. Chi, K.-H., No-Wook Park, C.J.C.: Fuzzy Logic Intergration for Landslide Hazard Mapping using Spatial Data from Boeun, Korea. In: Symposium on geospatial theory, processing and application, ottawa

A Parallel Nonnegative Tensor Factorization Algorithm for Mining Global Climate Data

Qiang Zhang[1], Michael W. Berry[2], Brian T. Lamb[2], and Tabitha Samuel[2]

[1] Department of Biostatistical Sciences, Wake Forest University Health Sciences, Medical Center Blvd, Winston Salem, NC 27157
qizhang@wfubmc.edu
[2] Department of Electrical Engineering and Computer Science, University of Tennessee, 203 Claxton Complex, Knoxville, TN 37996-3450
{berry,blamb,tsamuel}@eecs.utk.edu

Abstract. Increasingly large datasets acquired by NASA for global climate studies demand larger computation memory and higher CPU speed to mine out useful and revealing information. While boosting the CPU frequency is getting harder, clustering multiple lower performance computers thus becomes increasingly popular. This prompts a trend of parallelizing the existing algorithms and methods by mathematicians and computer scientists. In this paper, we take on the task of parallelizing the Nonnegative Tensor Factorization (NTF) method, with the purposes of distributing large datasets into each cluster node and thus reducing the demand on a single node, blocking and localizing the computation at the maximal degree, and finally minimizing the memory use for storing matrices or tensors by exploiting their structural relationships. Numerical experiments were performed on a NASA global sea surface temperature dataset and result factors were analyzed and discussed.

Keywords: nonnegative tensor factorization, parallel computation, data mining, global climate.

1 Introduction

Data mining techniques are commonly used for the discovery of *interesting* patterns in earth science data. Such patterns can help to both understand and predict changes in climate and the global carbon cycle. Regions of the earth can be partitioned into land and ocean areas from which subregions described by an ensemble land- or sea-based parameters are possible. Patterns within these subregions are mined to reveal both spatial and temporal autocorrelation. In this study, we sought to identify regions (or clusters) of the earth which have similar short- or long-term characteristics. Earth scientists are particularly interested in patterns that reflect deviations from normal seasonal variations (e.g., El Niño and La Niña). Interpreting these patterns can facilitate a better understanding of biosphere processes and the effects human policy decisions at a global scale. Such effects include deforestation, air and water quality, urbanization, and global warming.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 405–415, 2009.

Eigensystem-based analysis driven by principal component analysis (PCA) and the singular value decomposition (SVD) has been used to cluster climate indices [14]. Unfortunately, the orthogonal matrix factors (basis vectors) generated by the SVD are difficult to interpret and as discussed by Steinbach et al. in [13], stronger signals typically mask weaker signals. Among other data mining techniques, (approximate) Nonnegative Matrix Factorization (NMF) has attracted much attention since the early work of Paatero and Tapper [11] and Lee and Seung's seminal paper on learning the parts of objects [9]. In NMF, an $m \times n$ (nonnegative) mixed data matrix X is approximately factored into a product of two nonnegative rank-k matrices, with k small compared to m and n, $X \approx WH$. This factorization has the advantage that W and H can provide a physically realizable representation of the mixed data, due to the inherent nonnegativity constraint. Nonnegative Tensor Factorization (NTF) is a natural extension of NMF to higher dimensional data. In NTF, high-dimensional data, such as 3D or 4D global climate data, is factored directly and is approximated by a sum of rank-1 nonnegative tensors. See Figure 1 for an illustration of 3-D tensor factorization. Similar to NMF, we also see a quick development of NTF algorithms [12,15] and their applications in recent years. In this research, we exploit the nonnegative tensor factorization of multidimensional climate data in order to capture patterns/signals not possible with traditional 2-way factor analysis.

2 Parallel Nonnegative Tensor Factorization

In nonnegative tensor factorization (NTF), high-dimensional data, such as global sea surface temperature, is factored directly and is approximated by a sum of rank-1 nonnegative tensors. See Figure 1 for an illustration of a 3-D tensor factorization.

Definition 1. *Let* $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{D_1 \times D_2 \times D_3}$ *be a nonnegative tensor and define*

$$\hat{\boldsymbol{\mathcal{T}}} = \sum_{i=1}^{k} \boldsymbol{x}^{(i)} \circ \boldsymbol{y}^{(i)} \circ \boldsymbol{z}^{(i)},$$

to be in a CANDECOMP (CP) canonical factored form, where $\boldsymbol{x}^{(i)} \in \mathbb{R}^{D_1}, \boldsymbol{y}^{(i)} \in \mathbb{R}^{D_2}$, *and* $\boldsymbol{z}^{(i)} \in \mathbb{R}^{D_3}$ *are all nonnegative. Then, a rank-k nonnegative approximate tensor factorization problem is defined as*

$$\min_{\tilde{\boldsymbol{\mathcal{T}}}} ||\boldsymbol{\mathcal{T}} - \tilde{\boldsymbol{\mathcal{T}}}||_F^2, \text{ subject to } \tilde{\boldsymbol{\mathcal{T}}} \geq 0. \tag{1}$$

Given the large datasets we encounter with global climate data, our interest in this study is to parallelize the problem posed above and distribute computations evenly to processors in a distributed computing environment. By Definition (1), the NTF problem is posed as a non-linear optimization problem, which is not easily parallelizable. In a naive approach, we may separate the original data cube into 3D blocks and fit factors for each block in parallel. However, the factors from

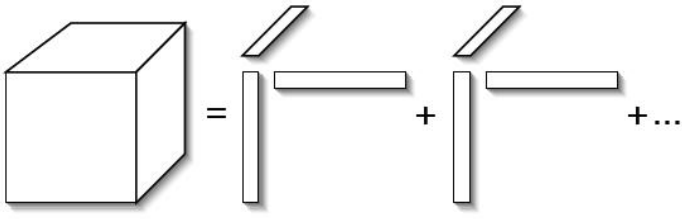


Fig. 1. An illustration of 3-D tensor approximate factorization using a sum of rank one tensors

each block may not match together, which is to say, blocks along each dimension should have an identical factor in either $\boldsymbol{X}$, $\boldsymbol{Y}$ or $\boldsymbol{Z}$. This almost certainly would not be the case when we optimize for three factors individually for each block.

Nevertheless, our hope lies in the fact that a common approach in solving (1) is the Alternating Least Squares (ALS) method [3,6], which is a special case of the block coordinate descent method, also known as the Block Gauss-Seidel (BGS) method [7]. At each iteration step, the BGS method (in alternating fashion) optimizes only a subset of the variables, while keeping the rest fixed, and turns the original non-convex problem into a sequence of convex least squares subproblems. In NTF, this means holding two matrix factors fixed while fitting for the other one. Thus, the original NTF problem is transformed into three semi-NMF (nonnegative matrix factorization) sub-problems in each iteration. Here we use the term "semi" to represent the optimization only for one of the two factor matrices, while assuming the other is given.

Definition 2. *Given* $\boldsymbol{A} \in \mathbb{R}^{m \times n} \geq 0$ *and* $\boldsymbol{W} \in \mathbb{R}^{m \times k} \geq 0$*, a semi-NMF problem is defined as*

$$\min_{\boldsymbol{H}} \Phi(\boldsymbol{H}) = ||\boldsymbol{A} - \boldsymbol{W}\boldsymbol{H}||_F^2, \ \textit{subject to } \boldsymbol{H} \geq 0. \tag{2}$$

One important observation on (2) is that solving (2) is equivalent to solving for each column of $\boldsymbol{H}$ independently, i.e.

$$\min_{\boldsymbol{h}_i} \Phi(\boldsymbol{h}_i) = \|\boldsymbol{a}_i - \boldsymbol{W}\boldsymbol{h}_i\|_F^2, , \tag{3}$$

where $\boldsymbol{a}_i$ and $\boldsymbol{h}_i$ are the column vectors of $\boldsymbol{A}$ and $\boldsymbol{H}$. This provides a great opportunity for parallelization of each semi-NMF subproblem, even though the original NTF problem is not defined for easy parallelization.

The ALS approach splits the NTF problem (1) into three semi-NMF subproblems, i.e. given $\boldsymbol{X}$ and $\boldsymbol{Y}$, we solve for $\boldsymbol{Z}$ by

$$\min_{\boldsymbol{Z}} \Phi(\boldsymbol{Z}) = \|\boldsymbol{T}_z - (\boldsymbol{X} \odot \boldsymbol{Y})\boldsymbol{Z}\|_F^2, \tag{4}$$

where $\boldsymbol{T}_z \in \mathbb{R}^{D_1 D_2 \times D_3}$ is the unfolded tensor $\boldsymbol{\mathcal{T}}$ along the z dimension and $(\boldsymbol{T}_z)_{(j-1)*D_2+i,k} = t_{ijk}$. $\boldsymbol{X} \odot \boldsymbol{Y}$ is the Khatri-Rao product of the two matrices. Next we fix $\boldsymbol{X}$ and $\boldsymbol{Z}$, and solve for $\boldsymbol{Y}$ by

$$\min_{\boldsymbol{Y}} \Phi(\boldsymbol{Y}) = \|\boldsymbol{T}_y - (\boldsymbol{X} \odot \boldsymbol{Z})\boldsymbol{Y}\|_F^2, \tag{5}$$

where $\boldsymbol{T}_y \in \mathbb{R}^{D_1 D_3 \times D_2}$ is the unfolded tensor $\boldsymbol{\mathcal{T}}$ along the y dimension and $(\boldsymbol{T}_y)_{(k-1)*D_3+i,j} = t_{ijk}$. Finally, we fix $\boldsymbol{Z}$ and $\boldsymbol{Y}$, and solve for $\boldsymbol{X}$ by

$$\min_{\boldsymbol{X}} \Phi(\boldsymbol{X}) = \|\boldsymbol{T}_x - (\boldsymbol{Z} \odot \boldsymbol{Y})\boldsymbol{X}\|_F^2, \tag{6}$$

where $\boldsymbol{T}_x \in \mathbb{R}^{D_2 D_3 \times D_1}$ is the unfolded tensor $\boldsymbol{\mathcal{T}}$ along the x dimension and $(\boldsymbol{T}_x)_{(k-1)*D_3+j,i} = t_{ijk}$.

Here we use a modified version of a Projected Gradient Descent (PGD) method developed by Lin [10] to solve the semi-NMF problem (2). The projected gradient descent method is basically adding a projection function to the regular gradient descent method.

$$\boldsymbol{H}^{(p+1)} = P_+[\boldsymbol{H}^{(p)} - \alpha_p \nabla \Phi(\boldsymbol{H}^{(p)})], \tag{7}$$

where the gradient is $\nabla \Phi(\boldsymbol{H}) = \boldsymbol{W}^T\boldsymbol{W}\boldsymbol{H} - \boldsymbol{W}^T\boldsymbol{A}$, and P_+ is the projection function onto the nonnegative domain. Lin [10] enhanced the performance of the PGD method by improving the search for the optimal step size using the Armijo rule.

Two observations can be made about the PGD method. First, to solve for $\boldsymbol{H}$, we only need to use two quadratic forms of $\boldsymbol{W}$ and $\boldsymbol{A}$, i.e. $\boldsymbol{W}^T\boldsymbol{W}$ and $\boldsymbol{W}^T\boldsymbol{A}$ and by comparing the sizes of two quadratic forms, i.e. $k \times k$ and $k \times n$, with the sizes of $\boldsymbol{W}$ and $\boldsymbol{A}$, i.e. $m \times k$ and $m \times n$, and knowing $m, n \gg k$, we can save considerable memory required to store these matrices. Second, we can also split $\boldsymbol{W}$ and $\boldsymbol{A}$ and compute $\boldsymbol{W}^T\boldsymbol{W}$ and $\boldsymbol{W}^T\boldsymbol{A}$ in parallel, i.e.

$$\boldsymbol{W}^T\boldsymbol{W} = \sum_{i=1}^{p} \boldsymbol{W}_i^T\boldsymbol{W}_i \text{ and } \boldsymbol{W}^T\boldsymbol{A} = \sum_{i=1}^{p} \boldsymbol{W}_i^T\boldsymbol{A}_i.$$

Here, $W_i \in \mathbb{R}^{m/p \times k}$ is a block sub-matrix of $\boldsymbol{W}$, $\boldsymbol{A}_i \in \mathbb{R}^{m/p \times n}$ is a block sub-matrix of $\boldsymbol{A}$, p is the number of processors, and $i = 1, \ldots, p$.

2.1 Distribution of Data

We distribute four matrices to independent processors in the following ways to facilitate parallel I/O and computation. We divide $\boldsymbol{T}_z$ by row blocks, $\boldsymbol{T}_{zi}$, each block having a size of $D_1 D_2/p \times D_3$. This allows for parallel loading of data. One important observation is that we do not need to save $\boldsymbol{T}_y$ and $\boldsymbol{T}_x$ in the memory due to their relationships with $\boldsymbol{T}_z$, stated in the following proposition (proof is straightforward and thus omitted). These relationships will be used in computing the quadratic forms.

Proposition 1. *The relationships between $\boldsymbol{T}_x$, $\boldsymbol{T}_y$ and $\boldsymbol{T}_z$ are:*

1. *Each column of $\boldsymbol{T}_y$ is a vectorized row block matrix of $\boldsymbol{T}_z$.*

2. *Each row block matrix of* $\boldsymbol{T}_x$ *is a transpose of the corresponding row block matrix of* $\boldsymbol{T}_z$.

We next divide $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$ by row blocks, each block having a size of $D_i/p \times k, i = 1, 2, 3$. This allows for parallel initialization and writing of these matrices to output. Note that if D_1, D_2 or D_3 cannot be divided by p, the last block will have a remainder number of rows.

2.2 Parallelization of the First Semi-NMF (4)

For convenience, we represent $\boldsymbol{X} \odot \boldsymbol{Y}$ by $\boldsymbol{W}$. We first collect $\boldsymbol{X}_i$ from each process(or) to form a full $\boldsymbol{X}$, and compute the local $\boldsymbol{W}_i = \boldsymbol{X} \odot \boldsymbol{Y}_i$. We then locally compute $\boldsymbol{W}_i^T \boldsymbol{W}_i$ and $\boldsymbol{W}_i^T \boldsymbol{T}_{zi}$ and compute their sums ($\boldsymbol{W}^T \boldsymbol{W}$ and $\boldsymbol{W}^T \boldsymbol{T}_z$, respectively) using the DSESUM2D subroutine provided by BLACS [4]. $\boldsymbol{W}^T \boldsymbol{T}_z$ is then partitioned into column blocks of a size $k \times D_3/p$ for input into the PGD subroutine. Thus, instances of the PGD subroutine run in parallel to solve for each block $\boldsymbol{Z}_i$. This process is illustrated in Figure 2.

2.3 Parallelization of the Second Semi-NMF (5)

Here, we represent $\boldsymbol{X} \odot \boldsymbol{Z}$ by $\boldsymbol{W}$. We first collect $\boldsymbol{Z}_i$ from each process(or) to form a complete $\boldsymbol{Z}$, and since we already have the complete $\boldsymbol{X}$ within each process(or), we can compute the complete $\boldsymbol{W} = \boldsymbol{X} \odot \boldsymbol{Z}$, and thereby compute $\boldsymbol{W}^T \boldsymbol{W}$. To compute $\boldsymbol{W}^T \boldsymbol{T}_{yi}$, notice that each column of $\boldsymbol{T}_y$ is a row block of $\boldsymbol{T}_z$,

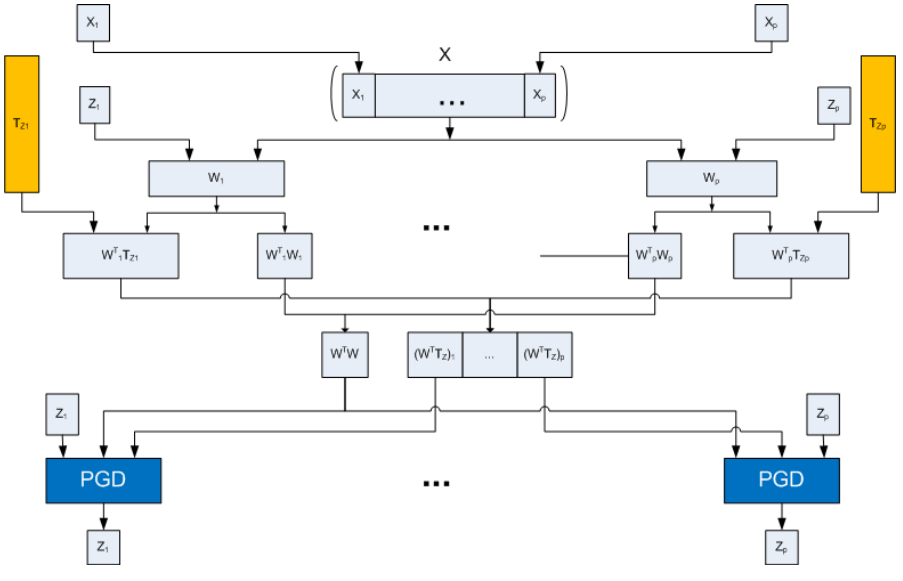


Fig. 2. Flowchart for the first semi-NMF subproblem (4) within NTF

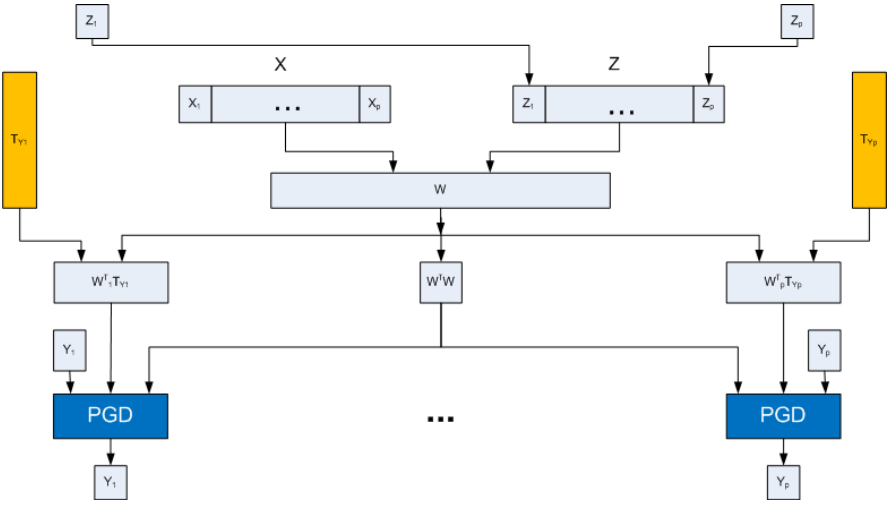


Fig. 3. Flowchart for the second semi-NMF subproblem (5) within NTF

and thus we can avoid saving $\boldsymbol{T}_y$ in memory and use the vectorized blocks of the local $\boldsymbol{T}_z$ to multiply with $\boldsymbol{W}$. Because $\boldsymbol{W}^T\boldsymbol{T}_{yi}$ is locally computed, it can used by a call to the PGD subroutine. Thus, independent calls to the PGD subroutine solve for each block $\boldsymbol{Y}_i$ in parallel. This process is illustrated in Figure 3.

2.4 Parallelization of the Third Semi-NMF (6)

Here, we represent $\boldsymbol{Z} \odot \boldsymbol{Y}$ by $\boldsymbol{W}$. We use the already collected complete $\boldsymbol{Z}$ and compute $\boldsymbol{Y}_i$ in order to formulate $\boldsymbol{W}_i = \boldsymbol{Z} \odot \boldsymbol{Y}_i$, which would then be used to compute $\boldsymbol{W}^T\boldsymbol{W}$. To compute $\boldsymbol{W}_i^T\boldsymbol{T}_{xi}$, notice that each row block of $\boldsymbol{T}_x$ is the transpose of the corresponding row block of $\boldsymbol{T}_z$, and thus we can avoid saving $\boldsymbol{T}_x$ in memory and use the row blocks of local $\boldsymbol{T}_z$ to multiply with $\boldsymbol{W}_i$. We deploy the LAPACK [1] subroutine DGEMM to avoid transposing $\boldsymbol{T}_{zi}$. To sum up $\boldsymbol{W}_i^T\boldsymbol{W}_i$ and $\boldsymbol{W}^T\boldsymbol{T}_{xi}$, we divide $\boldsymbol{W}^T\boldsymbol{T}_x$ into column blocks of a size $k \times D_1/p$ and make separate calls to the PGD subroutine. Again, these calls to PGD execute in parallel to solve for each block $\boldsymbol{X}_i$. The flowchart for this process is very similar to the one in Figure 2.

3 Data and Experimental Results

The six climate-based indices used for this study (see Table 1) were provided by researchers at the NASA Ames Research Center (ARC) in Moffett Field, CA. Pre-processing was performed to guarantee that the six variables matched the same coordinate system and time span. Most of the values are interpolated

Table 1. Climate variables considered in this study with adjustments (or shifts) to enforce nonnegativity (if needed)

Name	Description	Adjustment
sst	sea surface temperature	+273.15
ndvi	normalized difference vegetation index	+0.2
tem	land surface temperature	+273.15
pre	precipitation	
hg500	geopotential height (elevation) for barometric pressure of 500 millibars	+300
hg1000	geopotential height (elevation) for barometric pressure of 1000 millibars	+300

monthly averages on a uniform grid (with a slight distortion at the poles). Interpolation for some of the variables (such as geopotential height) necessarily produced negative values in some of the extreme regions (where it is difficult to sample or when surface pressure[1] is below 1000 mbar). It is not uncommon for many of the array values to be interpolated due to the sparsity of the original samples. The Arctic region, in particular, has few weather stations so that data values for many of the corresponding (lattitude, longitude) coordinates are interpolated from readings taken hundreds of miles away. Simple shifts (scalar increments) to these interpolated values are applied to all negative array elements to insure that all NTF input arrays are nonnegative.

Each parameter (from Table 1) is defined by a 3-way array or datacube of dimension $720 \times 360 \times 252$. The first two dimensions correspond to longitude and latitude coordinates, respectively, and the third dimension represents the month of reading. The time dimension spans from January 1982 to December 2002 for a total of 252 months (i.e., 21 years).

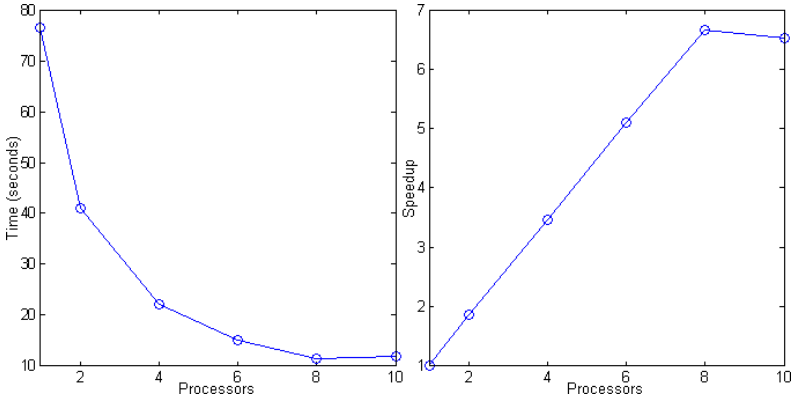


Fig. 4. Computation speedup of the parallel NTF algorithm

[1] Such was the case for the New Orleans area during hurricane Katrina.

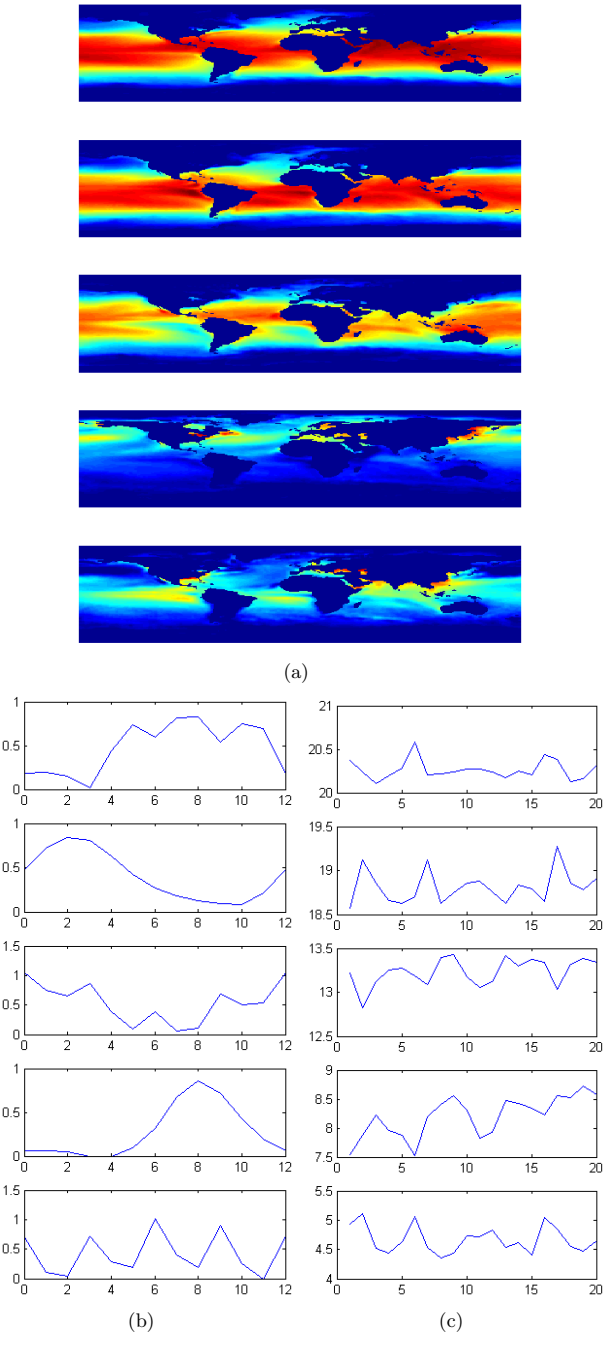


Fig. 5. (a) Global map of sea surface temperature patterns, (b) monthly variations of sea surface temperature patterns, (c) yearly variations of sea surface temperature patterns

All computations were performed on a Sun Fire X4600 M2 with 16 AMD 2.8GHz cores and 32GB of RAM. The original MATLAB®NTF codes were rewritten in C++ and compiled with several libraries including LAPACK [1], ScaLAPACK [2], BLACS [4], BLAS [5] and MPICH [8].

3.1 Speedup of the Parallel Computation

We use a simulated datacube in the size of $600 \times 400 \times 200$ to evaluate the speedup of the parallel NTF computation, by setting the number of processors at 1, 2, 4, 8 and 10. The size was deliberately chosen to be multiples of the number of processors to avoid any inconvenience in data distribution. The leftmost graph of Figure 4 shows the total computation time used for 100 iterations of the parallel NTF algorithm, taken from the processor that finishes last. The rightmost graph of Figure 4 shows the corresponding speedup. A sublinear speedup was achieved for 2 to 8 processors with an approximate peak speedup of 6.8 (among all runs with up to 10 processors).

3.2 Clustering Global Climate Data

The sea surface temperature parameter, originally in MATLAB®format, was partitioned into four sections and written in ASCII format for parallel reads by four different processes. The original data cube was first reshaped into a $259200 \times 12 \times 21$ array, and after removing sections corresponding to land-based coordinates the resulting data cube was $176876 \times 12 \times 20$. We also removed the last year data of data to make the time dimension a multiple of 4 for convenience.

Our intent was to extract typical monthly variation patterns in the second (tensor) factor, typical yearly variation patterns in the third factor and their corresponding global maps in the first factor. All three factors are shown in Figure 5, and they are sorted by the norm of the CP tensor from the greatest to the smallest in order to rank significance. We note that in Figure 5.b, the results in the last month are replicated at month 0 to reflect a full cycle.

The second factor represents El Niño, which has a characteristic peak in the winter and its global map shows a dark red tongue-shaped area off the coast of Ecuador. A yearly warming trend is observed in the fourth factor, mostly in the northern hemisphere and around the northern Pacific coastal area of China and Russia, and also to the northern Atlantic coastal area of the United States and Canada. It is also of interest to note that the last pattern shows some clear seasonal variations mostly along coastal areas (see the isolated red regions).

4 Conclusions

In this study, the nonnegative tensor factorization (NTF) method as a data mining tool is parallelized with the purpose of efficiently processing large datasets encountered in earth science. The parallel algorithm exploits the structural

relationships between matrices used in the original NTF algorithm for data distribution, memory savings and even computation task distribution. It is specifically applied to NASA-provided global land- and sea-based climate data with interesting results presented and analyzed for global sea surface temperature, in particular. Although not reported in this work, additional parallel NTF experiments using different combinations of the climate variables listed in Table 1 have been conducted. We expect to report on the results of clustering multiple land- and sea-based parameters in the near future.

Acknowledgement

This research was sponsored in part by the National Aeronautics and Space Administration (NASA) Ames Research Center under contract No. 07024004.

References

1. Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., Sorensen, D.: LAPACK Users' Guide, 3rd edn. SIAM, Philadelphia (1999)
2. Blackford, L.S., Choi, J., Cleary, A., D'Azevedo, E., Demmel, J., Dhillon, I., Dongarra, J., Hammarling, S., Henry, G., Petitet, A., Stanley, K., Walker, D., Whaley, R.C.: ScaLAPACK Users' Guide. SIAM, Philadelphia (1997)
3. Cichocki, A., Zdunek, R., Amari, S.: Hierarchical ALS Algorithms for Nonnegative Matrix and 3D Tensor Factorization. In: Davies, M.E., James, C.J., Abdallah, S.A., Plumbley, M.D. (eds.) ICA 2007. LNCS, vol. 4666, pp. 169–176. Springer, Heidelberg (2007)
4. Dongarra, J., Whaley, R.C.: LAPACK Working Note 94: A User"s Guide to the BLACS v1.0, Technical Report: UT-CS-95-281 (1995)
5. Dongarra, J., Du Croz, J., Duff, I.S., Hammarling, S.: A set of Level 3 Basic Linear Algebra Subprograms. ACM Trans. Math. Soft. 16, 1–17 (1990)
6. Faber, N.K.M., Bro, R., Hopke, P.K.: Recent developments in CANDECOMP/PARAFAC algorithms: a critical review. Chemometr. Intell. Lab. 65, 119–137 (2003)
7. Grippo, L., Sciandrone, M.: On the convergence of the block nonlinear Gauss-Seidel method under convex constraints. Operations Research Letters 26, 127–136 (2000)
8. Gropp, W., Lusk, E., Skjellum, A.: Using MPI: Portable Parallel Programming with the Message Passing Interface. MIT Press, Cambridge (1994)
9. Lee, D., Seung, H.: Learning the Parts of Objects by Non-Negative Matrix Factorization. Nature 401, 788–791 (1999)
10. Lin, C.: Projected gradient methods for non-negative matrix factorization. Neural Computation 19, 2756–2779 (2007)
11. Paatero, P., Tapper, U.: Positive matrix factorization a nonnegative factor model with optimal utilization of error-estimates of data value. Environmetrics 5, 111–126 (1994)
12. Shashua, A., Hazan, T.: Non-negative tensor factorization with applications to statistics and computer vision. In: Proceedings of the 22^{nd} International Conference on Machine Learning, pp. 792–799 (2005)

13. Steinbach, M., Tan, P.-N., Kumar, V., Klooster, S., Potter, C.: Discovery of climate indices using clustering. In: Proceedings of the Ninth ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2003), Washington, DC, August 24-27, pp. 446–455 (2003)
14. Storch, H.V., Zwiers, F.W.: Statistical Analysis in Climate Research. Cambridge University Press, Cambridge (1999)
15. Zhang, Q., Wang, H., Plemmons, R., Pauca, P.: Tensor methods for hyperspectral data analysis: a space object material identification study. Journal of Optical Society of America A 12, 3001–3012 (2008)

Querying for Feature Extraction and Visualization in Climate Modeling

C. Ryan Johnson[1], Markus Glatter[1], Wesley Kendall[1], Jian Huang[1], and Forrest Hoffman[2]

[1] University of Tennessee, Knoxville TN 37919, USA
[2] Oak Ridge National Laboratory, Oak Ridge TN 37831-6016, USA

Abstract. The ultimate goal of data visualization is to clearly portray features relevant to the problem being studied. This goal can be realized only if users can effectively communicate to the visualization software what features are of interest. To this end, we describe in this paper two query languages used by scientists to locate and visually emphasize relevant data in both space and time. These languages offer descriptive feedback and interactive refinement of query parameters, which are essential in any framework supporting queries of arbitrary complexity. We apply these languages to extract features of interest from climate model results and describe how they support rapid feature extraction from large datasets.

1 Introduction

Given the high-resolution and high-dimensionality of the datasets resulting from today's large-scale climate simulations, it is typically infeasible to visualize a dataset in its entirety. Moreover, the limits of hardware and human perception hinder real-time investigative analysis. Possible solutions to reduce the amount of visualized data may involve automatically detecting statistically unique locations [1] or using a problem solving environment supporting manual filtering of the data [2]. An alternative solution offering a balance between automation and control is to visualize or emphasize only a relevant subset of a dataset satisfying some query or hypothesis chosen by the user. To be effective, such queries must be expressed in terms of the problem domain and offer rich feedback, enabling interactive refinement of query parameters.

Modern fully-coupled general circulation models (GCMs) are frequently used to generate climate projections hundreds of years into the future. As spatial resolution and temporal output frequency increase, to better resolve and understand complex interacting phenomena, model output grows considerably. Analyzing this output through traditional means is becoming impossible. As a result, various data mining techniques, designed to extract features of interest and simplify very large time series datasets, are increasingly being applied to the climate modeling domain. Previous work by Hoffman *et al.* [3] has demonstrated the utility of such techniques by applying k-means cluster analysis to hundreds of years

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 416–425, 2009.

of climate model results. This paper describes additional techniques that show promise in extracting regional and temporal features in an automated fashion.

Herein we describe two query languages for visualizing features in any large, time-variant datasets. The first was initially developed for volume visualization in a previous work [4] to enable users to express features in terms of statistical properties of local spatio-temporal neighborhoods, while the second [5] is a temporal pattern language modeled after *regular expressions*, a powerful tool more commonly used for locating patterns in text. Both languages are tightly integrated into the visual analysis process. We apply these language frameworks to a recent climate modeling simulation and describe a mechanism making query processing scalable.

2 Neighborhood Distribution Querying

Gaining insight into high-dimensional climate data often requires investigating and testing statistical hypotheses. For instance, scientists may be curious about the differences in the interannual variability of snow coverage between two decades, whether or not precipitation and temperature are positively correlated, or if mean solar radiation is decreasing through time. Investigating these hypotheses at a global scale reveals only overall trends. Generally, it is more informative to examine local regions independently and draw conclusions from observations of smaller-scale phenomena. To this end, we describe a framework that enables statistical querying and visualization of spatio-temporal data. We develop a visual query language in which scientists can express arbitrary, statistic-based queries to determine which local regions meet a set of hypotheses.

2.1 Neighborhood-Based Querying

The core structure of our framework is the distribution of values in the local neighborhood around each data point. Scientists query for features of interest in terms of statistics derived from intervals within these distributions. Figure 1 illustrates the process of forming a query.

Neighborhood. The exact size and shape of the local region that forms the sample space can be customized by the user. By default, the neighborhood is defined spatially as the set of data points within a specified Euclidean distance of radius r. However, users can alter the shape to search for features more easily described in an anisotropic sample space. Additionally, neighborhoods may be defined across both space and time domains.

Bin Selection. With the neighborhood defined, the user selects which variables are used to define the feature to be visualized. The user may focus on a particular interval of values for each variable. We refer to the interval on which a variable is examined as a *bin*. In choosing a bin, a scientist narrows the distribution for which statistical measures are calculated to a relevant subset of the data. For example, an investigation of liquid water may involve only temperatures above freezing.

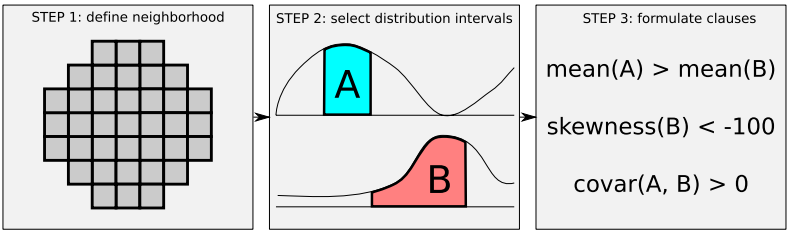


Fig. 1. An illustration of the statistical querying process. (a) Scientists first choose the spatio-temporal neighborhood serving as the sample space. (b) Next, the sample space is refined to intervals of the variables and their distributions relevant to the problem. (c) Lastly, scientists describe features of interest using inequalities relating the intervals' statistical primitives.

Querying by Predicate Clauses. With the sample space and distributions in place, queries are now expressed as a series of inequalities relating properties of distribution intervals to target criteria. The properties currently supported include relative frequency, mean, variance and standard deviation, skewness, and covariance. As an example predicate clause, let us consider the feature of rapid surface temperature change in the spatial domain. We select a circular neighborhood of radius 3, a bin encompassing all possible surface temperatures, and a predicate clause stating that the standard deviation must be greater than 3. We apply this query to the July 2000 timestep of the IPCC land-model simulation, with the result shown in Figure 2. Large elevational gradients are the primary feature emphasized by this query because these are the neighborhoods with large temperature deviations.

In the proceeding example, the target criterion is constant: standard deviation must be greater than 3.0 for a neighborhood to match. However, we can also allow target criteria to be expressed dynamically in terms of other interval properties. To query for locations where the mean percentage of a land grid square covered in snow one decade is half that of another decade, we select a neighborhood

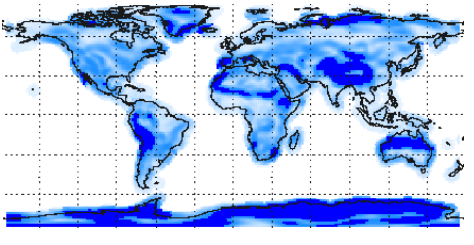


Fig. 2. A query for surface temperatures in circular neighborhoods of radius 3 in July 2000 having a standard deviation greater than 3°K. The dark locations fully match the query, while the lighter shade's opacity reflects how nearly a non-matching neighborhood came to matching.

spanning ten years, establish two intervals from 0% to 100% on snow coverage for each decade, and a clause requiring `mean(bin 1) < 0.5 * mean(bin 0)`.

Additionally, features typically involve compound criteria, and our query language readily handles multiple predicate clauses. Neighborhoods may match only a subset of the query's clauses, and in order to visualize partial matches, we record a *predicate signature*, which is a bitfield with bit i set to 1 if the criterion of clause i is fully met in a neighborhood. A neighborhood's predicate signature is used to determine its color when visualized. Each clause may also be assigned a weight reflecting its degree of importance in defining a feature.

To avoid a misleading boundary between matching and non-matching neighborhoods, each neighborhood is also assigned a score in $[0, 1]$ indicating how closely the criteria of the clauses are met. The score for an individual clause is assigned according to an exponential decay function of a neighborhood's distance from the target criterion and a dropoff parameter that controls how quickly the score drops with distance, as detailed in Equation (2). A neighborhood's total score is defined in Equation (3) as a weighted sum of the clause scores.

$$\Delta = \begin{cases} 0 & \text{if criterion is met} \\ |lhs - rhs| & \text{otherwise} \end{cases} \tag{1}$$

$$score_i = \exp\left(\frac{\Delta^2 \times \ln(.5)}{dropoff_i^2}\right) \tag{2}$$

$$score = \sum weight_i \times score_i \tag{3}$$

2.2 Query Visualization

A query is visualized by coloring each location with its neighborhood's predicate signature, which represents the combination of clauses that are fully met. Scientists can interactively customize the colormap indexed by these predicate signatures or make a particular combination completely transparent. The neighborhood's score is used to increase the neighborhood's transparency further. By default, low-scoring neighborhoods are assigned a transparency close to 0, while high-scoring neighborhoods are more opaque.

Figure 3 is an example visualization of a query with multiple clauses. The query investigates the relationship between mean temperature increase and mean precipitation between the first and last decades of the IPCC dataset. Each color represents a different combination of the three criteria being met, though only six of the eight possible interactions actually occur.

2.3 Implementation

To formulate distribution queries for spatio-temporal neighborhoods, we have built a graphical tool that guides the user in selecting neighborhood sizes, bin ranges, and criteria. Changes to query parameters instantaneously update the visualization for immediate and interactive feedback. Using the mouse, users can hover over each pixel and see exactly which clauses are fully met without having

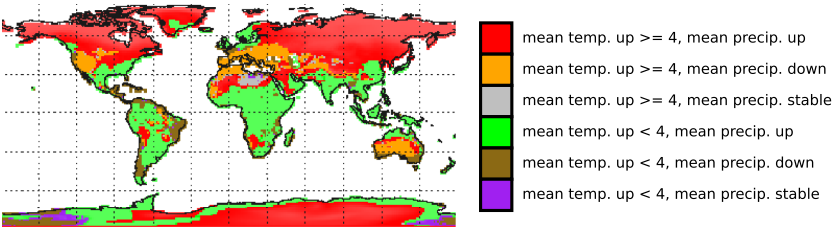


Fig. 3. An investigation of mean temperature and precipitation between decades 2000–2009 and 2090–2099. Six possible interactions are observed in the simulation: mean temperature may or may not go up 4°K or more, and mean precipitation may increase, decrease, or stay constant. Each possible combination of events yields a unique predicate signature and color.

to decipher the colormap. A separate pane allows each individual clause to be studied in isolation from the entire query. The visually-composed neighborhood, bin, and clause information can be saved in an XML format and reloaded and modified as needed.

```
for each neighborhood h
   for each neighbor n
      for each variable v to be binned
         if n.v in bin
            add n.v to bin
   calculate bin statistics
   for each clause c
      if c.criteria met
         c.score = 1
         set bit in h.predicate_signature
      else
         calculate c.score according to dropoff
      h.score = h.score + c.weight * c.score
```

Fig. 4. The algorithm for evaluating a neighborhood distribution query. The output of this algorithm is a set of predicate signatures and scores for each neighborhood in the dataset.

3 Temporal Querying

Using query-based visualization, users can sift through very large and highly complex multivariate data. Though queries are often guided by human-domain knowledge, query-based visualization commonly involves trial and error. Unfortunately, this approach typically does not scale as datasets grow exponentially large. In this work, we describe a query language for accelerating discovery of temporal connections between multivariate patterns of interest in climate modeling simulation data.

3.1 Textual Pattern Matching

Motivated by the elegance and power of regular expressions and globbing in text string searching, we have developed a text-based search language for concisely specifying temporal patterns. In our system, a user hypothesis can be loosely defined. In traditional regular expressions, wildcards are used to support inexact matching. For example, `file*.pdf` is an expression that matches any files named with prefix `file` and extension `pdf`. Our system accepts a similar kind of qualitative query. We use wildcards to provide a powerful way of representing the existence of temporal events.

To application scientists, this method of vaguely specifying temporal patterns to visualize is extremely useful. Domain knowledge is often expressed in a qualitative manner, and scientists can be hard-pressed to define exact data queries to extract meaningful subsets of the data. Using our temporal regular expression language, qualitative patterns containing wildcard characters can be entered and expanded to a set of discrete data queries that are extracted from the dataset. For a scientist, this offers a more natural way of entering qualitative domain knowledge and avoids a potentially lengthy search process guided only by trial and error.

3.2 Pattern Matching and Syntax

For querying, we employ range queries, a widely used method for selecting subsets of multivariate data. Even though range queries are usually discrete, our system accepts quantitative queries that match and support a user's unclear or imprecise understanding of data. Thus, we allow a user to issue "fuzzy queries" in order to explore a data set he is not highly familiar with. Range queries may contain wildcard characters such as `*` and `?` that are expanded to generate actual range queries, much like the UNIX function `glob()` expands wildcards in `file??.pdf` and regular expressions expand patterns such as `file.*\.pdf`.

However, our language has a few non-traditional elements. The first one is `T`, the temporal mark. A search of `[4]*T[5]*` means that we are looking for patterns where an attribute is valued at 4 for zero or more timesteps and then changes to value 5. The temporal mark is the time of this event's occurrence, and in this case it is chosen to be the instant of the first timestep of the value 5. Our parser extracts the `T` from the expression and then generates all discrete patterns of interest. The location of `T` is recorded so as to indicate the precise time of the event's occurrence in further visualizations or data explorations.

The following list of examples demonstrates accepted wildcards, special characters, and valid data ranges:

- `[-1e15 - 1025.7]` – data ranges for a single timestep. This range contains a from-value and a to-value. `[74.2]` has both values being identical (same as `[74.2 - 74.2]`).
- `[0.3 - 10e9]*` – a data range applied to zero or more sequential timesteps.
- `?`, `?*` – wildcard ranges. The first represents the entire range of data values for a single timestep. The second represents the entire range of data values for zero or more timesteps.

- `T` – the (optional) temporal mark. It marks the time index at which the event that is subject of the query occurs.
- `[1 - 5]*[8]` – a query without a temporal mark is also valid. This query addresses all items for which values in all timesteps are between 1 and 5, and a value of 8 in the last timestep.

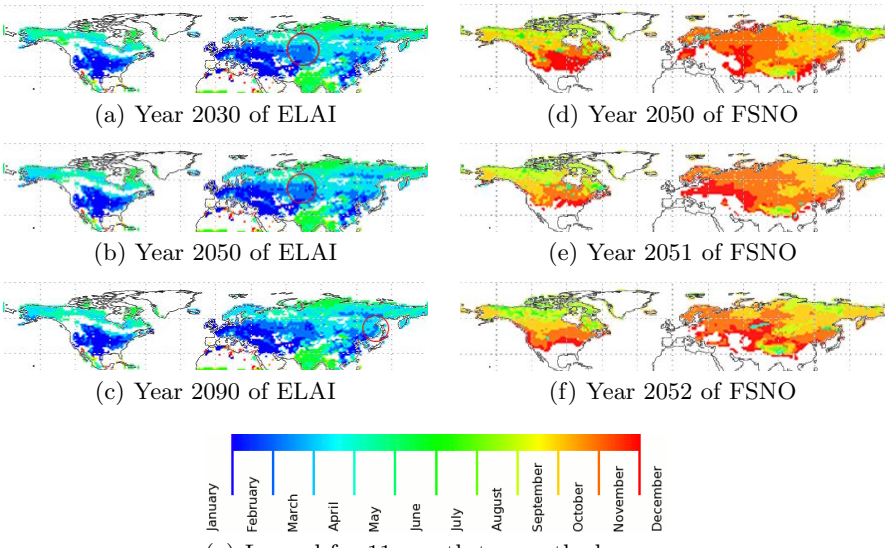
3.3 Querying Global Climate Model Results

Using the results from a global climate model, we establish "events" that are defined by one or more variables changing over time at different spatial locations. We mark the time of such events at each spatial location with the temporal mark `T`, and color individual pixels according to the extracted times. As we are considering temporal change between consecutive months, color indicates the point in time in between the months in which the event occurred, *e.g.*, given the colormap in Figure 5(g), a blue pixel depicts that the event happened in the transition from January to February.

In Figures 5(a) through 5(c), we display the temporal change of the variable ELAI (Exposed one-sided leaf area index in units of m^2 of leaf area per m^2 of ground area) over the course of one year. We query for data locations that experience a positive relative change of more than 40% after a period of zero or more timesteps in which the relative change is low (between −40% and +40%). The purpose of this query is to find the beginning of the growing season in the Northern Hemisphere. We mark the first timestep (month) in which the threshold of change is exceeded and color the pixels accordingly. We chose 40% as the threshold for spring green-up because smaller changes in leaf area index do not represent a temporal feature of interest.

We can immediately see how the event of marked temporal change in ELAI follows a certain path as the year progresses. As we expect, it begins in the southern parts of North and Central America and generally advances north month by month. On the Eurasian landmass, a similar progression to the north is visible, as is a progression from central Europe towards central Russia and Siberia. We also notice that the differences between years are minor and almost imperceptible. The temporal change of the variable ELAI over the course of each year appears to be very stable.

Figures 5(d)-(f) display the results of a query designed to identify the point in time when the first large snowfall between May and December occurs. The query then considers only data locations with snow cover (variable FSNO, representing the fraction of ground covered by snow) is either reduced or increased by no more than 7%. When we encounter the first temporal change of the snow fraction larger than 7%, we consider it the first large snowfall and mark its timestep (month). We chose 7% since it gives us a good threshold that makes our query resistant to minor changes of the snow fraction which would generate a false and potentially early temporal mark. Again, we can clearly make out the underlying pattern. The snow cover first grows larger by more than 7% in northern Canada and Siberia, as well as the Himalayas, and then progresses into the warmer regions to the south and, in the case of the Eurasian landmass, to the west. One can also recognize the Rocky Mountains in the Western U.S. as an area of early snowfall.



(a) Year 2030 of ELAI (d) Year 2050 of FSNO

(b) Year 2050 of ELAI (e) Year 2051 of FSNO

(c) Year 2090 of ELAI (f) Year 2052 of FSNO

(g) Legend for 11 month-to-month changes

Fig. 5. (a-c). A query for significant change in ELAI in the northern hemisphere. Locations are colored according to their temporal mark `T`, which indicates the time at which the change occurs. The colormap is shown in (g). Areas highlighted in red circles indicate locations where the temporal marker shifts most between decades. (d-f) A query for the time of the first large snowfall in the northern hemisphere.

3.4 Query Performance

Our system for evaluating queries is implemented using scalable data servers described in Section 4. All timing results have been collected using an AMD 2.6 GHz Opteron cluster connected by InfiniBand DDR 4X network and the climate data set from section 3.3. Measured performance metrics of our system are highly dependent on the data and the query. With most tests that we have run using 20 compute nodes as data servers, the query performance is acceptable for interactive use.

4 Scalable Querying

With the ever-increasing size of scientific simulation data, fast querying requires a scalable solution to managing and extracting features of the data. Previous methods have demonstrated scalable querying of large multivariate data [6], but I/O and preprocessing times have not been considered. However, for *in situ* querying of simulation data—in which the data is examined as it is created—these I/O and preprocessing times play a crucial role in the user experience. We describe a method of leveraging the power of modern supercomputing resources and parallel I/O to make this data-intensive query processing as fast as possible.

Table 1. Performance results on 20 compute servers. Running time is the wall clock time between query invocation and receipt of all matched locations.

Query	# Locations matched	Running time (secs)
`[0-10][0]?[0-1e10]*`	3	3.168
`[0-99][0]?[0-1e10]*`	3	26.522
`[40-60]??[0-1e10]*`	342	6.067
`[50]??*?*`	3,615,888	7.454
`[70]?[-5-1e10]*[5--1e10]*`	6,584	7.485
`[0-20]??[0-1e10]*?*`	3,593,696	159.97
`[80]?[-100-1e10]*[100--1e10]*[-100-1e10]*`	16,994,091	248.87
`[80-82]?[-100-1e10]*[100--1e10]*[-100-1e10]*`	50,565,859	757
`[0-99]?[-100-1e10]*[100--1e10]*[-100-1e10]*`	$\approx 1,685,529,000$	$\approx 72,500$

With the present ability to do calculations at one petaflop and beyond, many applications are bound by I/O speeds, and users of high performance systems realize that the memory access rate often determines the performance of their applications [7]. To reduce I/O bottlenecks, we employ existing parallel I/O libraries such as Parallel netCDF [8] for high-performance access to scientific datasets. Scientists create a configuration file of all the variables of interest from an arbitrary number of files. The configuration file is passed on to a routine that encapsulates the task of loading the data from disk using all available processors. This routine achieves maximum bandwidth by dividing the loading of data across processors and performing collective I/O when possible.

After being read from disk, the data must be distributed to keep the query processing load-balanced. In previous work [6], we have shown a method that offers near-optimal load-balancing among servers. Both query languages described above operate on value ranges of the data, and accordingly, processing a query can be accelerated by first sorting the data. The entire dataset is sorted in Hilbert space and distributed to the servers. To provide scalable sorting, we use an algorithm that keeps the data distributed among all the processors throughout the entire sort. This algorithm performs a global merge of the data and then swaps the data in a round-robin fashion. With the data quickly narrowed down to only relevant intervals, the rest of the query criteria can be evaluated in a distributed manner.

5 Conclusion

We have described two query languages for extracting features from very large scientific datasets, and have demonstrated their utility by applying them to extract and visualize features of interest from climate model output. Such tools are becoming increasingly important as simulations generate higher spatial and temporal resolution datasets that necessitate use of novel techniques for analysis. Neighborhood-based querying is useful for extracting spatial patterns from large datasets based on the statistical parameters of a region's frequency distribution. Temporal querying is useful for extracting patterns of change from very large

time-variant datasets. High performance I/O and parallel data processing are technologies that enable interactive query evaluation.

References

1. Jänicke, H., Wiebel, A., Scheuermann, G., Kollmann, W.: Multifield visualization using local statistical complexity. IEEE Transactions on Visualization and Computer Graphics 13, 1384–1391 (2007)
2. Kehrer, J., Ladstädter, F., Muigg, P., Doleisch, H., Steiner, A., Hauser, H.: Hypothesis generation in climate research with interactive visual data exploration. IEEE Transactions on Visualization and Computer Graphics 14(6), 1579–1586 (2008)
3. Hoffman, F.M., Hargrove, W.W., Erickson, D.J., Oglesby, R.J.: Using clustered climate regimes to analyze and compare predictions from fully coupled general circulation models. Earth Interactions 9(10), 1–27 (2005)
4. Johnson, C.R., Huang, J.: Distribution driven visualization of volume data. IEEE Transactions on Visualization and Computer Graphics (2009) (to appear)
5. Glatter, M., Huang, J., Ahern, S., Daniel, J., Lu, A.: Visualizing temporal patterns in large multivariate data using textual pattern matching. IEEE Transactions on Visualization and Computer Graphics 14(6), 1467–1474 (2008)
6. Glatter, M., Mollenhour, C., Huang, J., Gao, J.: Scalable data servers for large multivariate volume visualization. IEEE Transactions on Visualization and Computer Graphics 12(5), 1291–1298 (2006)
7. Ross, R., Peterka, T., Shen, H.-W., Ma, K.-L., Yu, H., Moreland, K.: Visualization and parallel I/O at extreme scale. Journal of Physics (Conference Series) 125 (July 2008)
8. Li, J., Liao, W.K., Choudhary, A., Ross, R., Thakur, R., Gropp, W., Latham, R., Siegel, A., Gallagher, B., Zingale, M.: Parallel netCDF: A high-performance scientific I/O interface. In: Proceedings of Supercomputing (November 2003)

Applying Wavelet and Fourier Transform Analysis to Large Geophysical Datasets

Bjørn-Gustaf J. Brooks

Iowa State University, Ames IA 50011, USA
bjorn@climatemodeling.org

Abstract. The recurrence of periodic environmental states is important to many systems of study, and particularly to the life cycles of plants and animals. Periodicity in parameters that are important to life, such as precipitation, are important to understanding environmental impacts, and changes to their intensity and duration can have far reaching impacts. To keep pace with the rapid expansion of Earth science datasets, efficient data mining techniques are required. This paper presents an automated method for Discrete Fourier transform (DFT) and wavelet analysis capable of rapidly identifying changes in the intensity of seasonal, annual, or interannual events. Spectral analysis is used to diagnose model behavior, and locate land surface cells that show shifting cycle intensity, which could be used as an indicator of climate shift. The strengths and limitations of DFT and wavelet spectral analysis are also explored. Example routines in `Octave/Matlab` and `IDL` are provided.

1 Introduction

Many geophysical systems under study today are seasonal or operate on cycles in which particular states are revisited. Variation in systems that show recurrent states (*e.g.* climate change, the El Niño Southern Oscillation, Milankovitch cycles) is one of the most important topics in Earth science today. There exist a variety of mathematical methods for determining their dominant modes, among which Fourier and wavelet analysis both have the advantage of having common libraries and routines in several languages and software packages that can be used to develop automated routines to rapidly search for changes to the intensity and periodicity of cycles.

This paper uses an example dataset of temperature, precipitation, and soil moisture model output from the Parallel Climate Model (PCM). PCM is a fully coupled model. Ensembles of simulations were used to show global mean surface temperature increase of $\sim$1.9 °C during the twenty-first century and a 3% increase in global precipitation[1]. A more complete description can be found in [2]. Figure 1 represents the 10 year running average of the 2796 land surface grid cells clustered into ecoregions by Hoffman and others[1], who used the descriptive variables of temperature, precipitation, and soil moisture to assign each land

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 426–434, 2009.

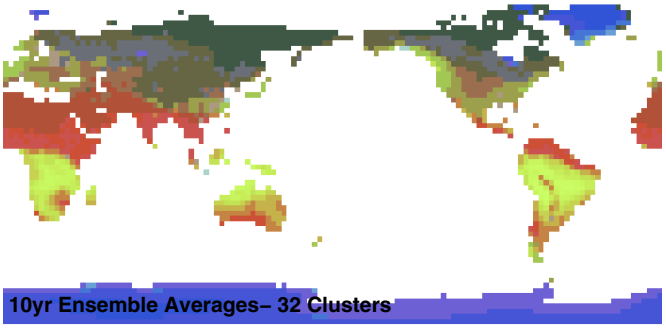


Fig. 1. Global map of 2796 PCM land surface cells taken from Hoffman and others[1]

surface cell to 1 of 32 ecoregion types through a 99 year (1188 month) Business As Usual (BAU) general circulation model (*i.e.* PCM). Three fields of ecological importance were analyzed: temperature (K), precipitation ($\mathrm{kg\,m^{-2}\,s^{-1}}$), and soil moisture (fractional volume of root zone water ranging from 0:1).

The colors of each land surface cell in Figure 1 indicate the ecoregion type assigned by the Multivariate Spatio-Temporal Clustering (MSTC) procedure[1; 3], which was based on an iterative k-means algorithm[4]. Each land cell at each time point is described by three parameters of temperature, precipitation, and soil moisture that define its location in three-dimensional data space. Hoffman and others grouped these data points into clusters, called ecoregions, based on their euclidean distances. Climate changes for a land cell were said to have occurred when a land cell entered a climate regime that it had never previously visited.

To suit the purposes of this paper a scalable DFT routine was employed to identify particular PCM land surface cells that showed significant changes to the periodicity of their annual cycles. Two example land surface cells were selected for detailed wavelet analysis and to demonstrate how changes to the annual cycle within a time series of interest are located. The primary focus is to identify such variant land surface cells, which may serve as potential indicators of climate change.

2 Discrete Fourier Transform Signal Detection

Discrete Fourier transform analysis can be a useful technique for detecting periodic signals in time series[5]. Periodic signals within a time series are obtained by decomposing the time series into two parts, a real, $\Re\{F_n(x)\}$ and an imaginary, $\Im\{F_n(x)\}$. The power spectra of a DFT (Figure 2A) can be used to represent the cumulative magnitude of a signal within a time series. Changes to the spectral power of peaks within different window segments of a time series (*e.g.* a slackening annual cycle) can be identified by segmenting the time series into windows and computing the DFT power spectrum (DFTPS) for each window. When combined with Monte Carlo simulations, estimates for the significances of peaks in a power spectrum can also be determined. Example routines are available online at: www.climatemodeling.org/pcm/cyclicity/.

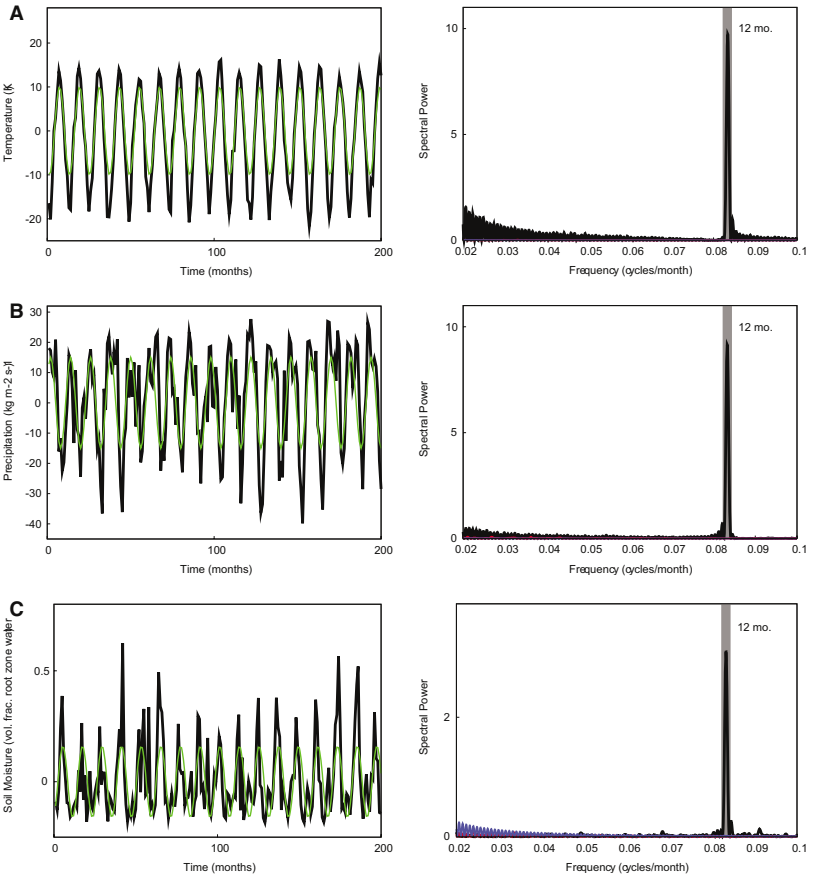


Fig. 2. Comparison of Discrete Fourier transform power spectrum (DFTPS) and time series plots. The DFTPS of three time series (mean subtracted) $\{(t_i, x_i)\}_{i=1}^{200}$ (soil moisture, precipitation, and temperature) of one Middle Eastern land cell from the Parallel Climate Model are shown in the left column with their corresponding DFT power spectra. Gray bars in the DFTPS locate the annual (12 month) period. Row A. represents temperature, B. precipitation, and C. soil moisture.

To compute the DFTPS in the second column of Figure 2, a 200 month segment of the original time series was extracted, $\{(t_i, x_i)\}_{i=1}^{200}$, which represents PCM precipitation data sampled at monthly increments x_i. The series was standardized to a variance of 1, and then expressed as the sum of a series of sines and cosines through Fourier expansion. In its simplest form the DFT is denoted by Equation 1

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi nk/N}, n = 0, 1, 2 \ldots N-1 \tag{1}$$

procedure pcm_locate
input: time series, $\{(t_i, x_i)\}_{i=1}^{nterms}$
output: a_k (amplitude), and tau_k

```
for k = 1,nw do
   w = x[i:i+ne]
   w_n = w/std_dev(w)
   w_ft = fft(w_n)
   w_sp = w_ft * conj(w_ft)
   a_k = real(w_sp)
   tau_k = imaginary(w_sp)
end do
```

Fig. 3. The general procedure that locates windows within a time series with significant variation in the period and amplitude of the annual cycle. `nw` is the number of windows within the time series. `ne` is the number of elements (`x`) within each window.

where $x(n)$ is the input time series at sample n, and $j = \sqrt{-1}$ is the basis for complex numbers. Many software packages have built in libraries and routines for DFT analysis and for producing DFTPS.

Figure 2 represents the monthly PCM output of temperature, precipitation, and soil moisture for one land surface cell. The left column represents the time series of the first 200 months of temperature, precipitation, and soil moisture for that same grid cell. The right column of DFT power spectra show the DFTPS of cycles within the three time series. The DFTPS each have one dominant cycle near $f = 0.083$ (12 months), and can be used to answer two types of questions: 'What are the periods of cycles in the time series' and 'What are their significances?'

A well known limitation of Fourier expansion is that it has no time resolution meaning that although DFTPS indicate the frequencies present in a signal, nothing is indicated about where they are present in that signal or how their intensity changes through time. To avoid such problems a routine a routine was developed (pcm_locate.m) that divides each time series into window segments and computes their DFTPS to search for changes to the intensity and duration of the annual annual cycle (Figure 3). The results of all the land surface cells were sorted to identify particular cells that show dramatic change in their annual cycle. This routine can be scaled to run on a cluster.

3 Wavelets

Unlike Fourier expansion, wavelet functions broaden time series analysis into time-scale space, which allows wavelet methods to detect periodicities that are intermittent throughout the dataset[6]. Wavelet transform slides a window along

the signal calculating the spectrum at each position, solving the time-domain limitation of Fourier expansion by a method more efficient that windowing[7]. Wavelet spectrograms can be used as representations of time and frequency-domain changes for the variant land surface cells once identified.

Wavelet functions (ψ) concern the relationship between a dilation parameter S, also called scale, and a translation parameter τ, also called shift and are described by Equation 2.

$$\Psi_{S,\tau}(\eta) = \frac{1}{\sqrt{S}}\Psi\left(\frac{\eta - \tau}{S}\right) \tag{2}$$

η is a nondimensional sampling parameter, and $S^{-1/2}$ normalizes the energy across different scales, which allows the spectral power of a signal to be compared across different scales. Equation 2 shows a key distinction of wavelet analysis in that it does not specify a wavelet basis function. This flexibility permits different wavelet functions to be used. Changing the wavelet function or varying the parameters that accompany the wavelet function, for example the scale, has important effects on the resulting spectrogram. Although many studies use the Morlet wavelet function for its frequency resolution, in this paper the Paul function is used to provide better time resolution. A comparison of the uses of different wavelet functions can be found in [8].

Wavelet basis functions or wavelets have zero means (Figure 4) and are localized in time and Fourier space. Given an array of equally spaced observations, x_n, from $n = 0 \ldots N-1$, a continuous wavelet function, $\psi(\eta)$, can detect variations in the power of an array of data[9].

As in Fourier analysis the resulting wavelet power spectrum is partitioned into a real part (the amplitude) and an imaginary part (the phase information). The key distinction of wavelet transform is that the period and spectral power of the wavelet transform can be plotted through time (Figure 5). Strongly cyclic events are indicated by hot colors on the spectrogram, while small peaks and

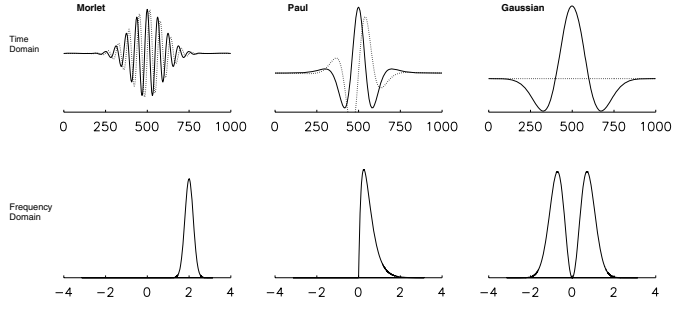


Fig. 4. Wavelet functions. The plots in the upper row indicate the real (solid) and imaginary (dotted) parts in the time domain, while the bottom row corresponds to the frequency domain of those same wavelet functions.

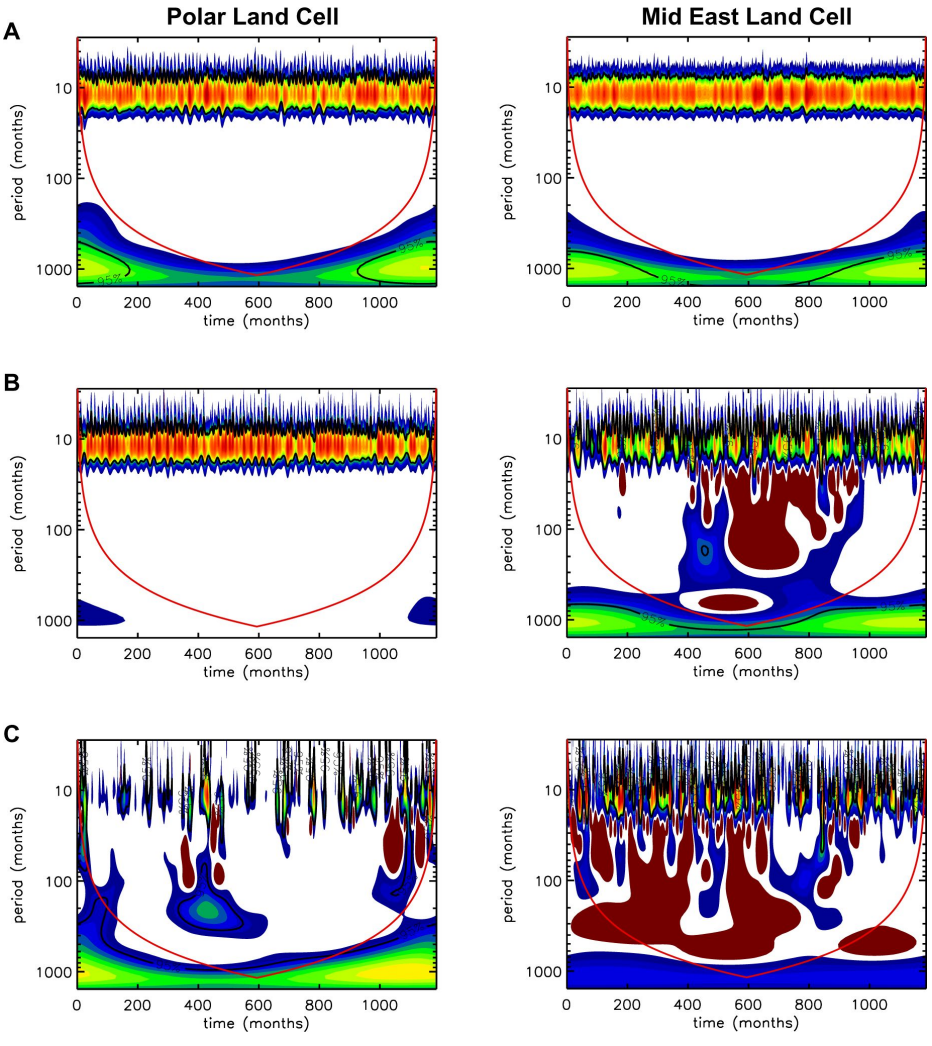


Fig. 5. Paul wavelet spectrograms of two land surface cells, divided into left (a typical polar cell) and right (a particularly variant Mid East cell) columns. Hot colors (red) indicate high spectral power (a strong cycle), whereas blue indicates little spectral intensity (a weak cycle). A. represents the temperature data for both cells. B. the precipitation, and C. soil moisture. The cone of influence is shown by the red curve, and a 95% confidence interval is indicated for the annual cycle by the black line encircling it.

weak cycles are indicated by cool colors. These plots can be reproduced using the pcm_wv.pro script in the online repository.

Because a continuous wavelet transform of a finite length time series has edge artifacts near the beginning and end of the power spectrum that represent an incomplete localization of the data, a cone of influence (COI) is used in wavelet

spectrograms to represent the threshold below which the significance of cycles is dubious[6]. However, if dealing with geographic data, such as for a band of longitude with fixed latitude, the data are continuous (*i.e.* they wrap around) and no COI is necessary.

4 Monte Carlo Statistical Significances

Although a spectral peak may appear in the data, it could be a computational artifact. Monte Carlo (MC) simulations can be used as null models to determine the statistical significances of peaks within a power spectrum. For any time series there are frequencies within which peaks are likely to be the result of random chance. MC trials are one way to decipher the significance of spectral peaks from random chance. Two MC routines are used as null models and the significance of spectral peaks in the original signal are computed by counting the number of times the amplitude of a peak generated by a random permutation of the data is able to exceed that of the original time series.

As opposed to purely theoretical (white noise) simulations, the two MC methods used here simulate the probability of occurrence of spectral peaks based on assumptions about the cause of the periodicity. These MC methods are adapted from Cornette[10], and Rohde and Muller[11]. They assume that fluctuations in the DFTPS follow an exponential form, such that the probability of spectral power at frequency f would be $P(h) = e^{-h/h(f)}$ where $h(f)$ denotes the average height h of the power spectrum at frequency f across many trial simulations.

In the first method, random walk trials, at each time increment t_i, a data point x_i is randomly selected from the series and used to generate a permutation of the original sequence $\{(t_i, x_i)\}_{i=1}^{1188}$. This represents the null hypothesis that diversity is a random walk. It is important to reject that variations in the time series are not best described by a random walk for cases where the fundamental dynamics of a system are complex or poorly constrained. The plot of the mean of 10,000 random walk trials is shown by the red curve in the right column of Figure 2.

For random block trials, the time series data are partitioned into N blocks of length M. At each increment of N the blocks are randomly permuted to obtain a random sequence. This method preserves some short term relationships between data points in the time series, and represents the hypothesis that variation in the time series is independently driven (*i.e.* directed) with major random perturbations. For the temperature, precipitation, and soil moisture PCM output a block length, M, of 12 (months) is used to represent the null model that seasonal trends are directed, but interannual variability is random. This is represented by the blue curve that appears in the right column of Figure 2. An example script for producing this plot can be found in our online repository (dftps.m).

Significance is computed for a peak of height h at a frequency f, as the fraction, p, of the randomly generated sequences for which the MC spectrum at f exceeds h. Table 1 lists the significance of the annual cycle in precipitation for the Middle Eastern cell in three different windows of the time series to show its evolution (*cf.* Figure 5B). The probability that a MC simulation of the 12 month

Table 1. The significances of the annual peak in the data from Figure 5B (precipitation data for the Mid East cell) fluctuates throughout the time series. Statistical significances of the 12 month cycle relative to 10,000 random walk (RW) and random block (RB) Monte Carlo simulations are given below in four conterminous 48 month windows of the time series between 600 and 743 months.

window segment	location in time series	RW significance	RB significance
	precipitation		
One	600-647	0.002	0.001
Two	648-695	0.003	0.000
Three	696-743	0.043	0.032

cycle has a greater significance is very small ($p < 0.005$) in windows one and two. Segment three however, has a reduced significance that seems to indicate a slackened annual cycle near 700 months.

5 Summary of Results

Reliable climate change determinations require powerful analytical tools, among which spectral analysis can be included. Figure 1 shows an example output from Hoffman and others[1] MSTC technique, which assigned each of the 2796 land cells to one of 32 climate regimes at each time point (month). Their evolution was tracked through time. Hoffman and others chose one representative Middle Eastern cell as an example that underwent a climate change by transitioning into the hottest and driest climate state, a regime it had never previously visited. The rigorousness of this climate transition can be explored by showing a shift in the intensity and duration of its annual cycles, for example in the duration of wet season precipitation events or in the intensity of peak summer temperatures.

While, temperature spectrograms for the polar and Middle East land cells are nearly identical (Figure 5A), substantial differences exist in soil moisture and precipitation. The first 600 months of soil moisture data for the Middle East land cell contrasts the last 600 months in Figure 5C. The first half is characterized by a strong annual soil moisture cycle, which are accompanied by a range of possibly significant $12-120$ month interannual soil moisture cycles. The second half of the time series shows noticeably fewer of these interannual cycles and more frequent gaps in the annual soil moisture cycle. The low frequency (red and blue) cycles that appear below the annual period should be regarded with caution, as they have a limited recurrence and do not have statistical significances equal to that of the 12 month cycle.

A diminishing annual soil moisture cycle could be the result of two phenomena. It may be the consequence of changes in air temperature or precipitation. Little change appears throughout the temperature spectrogram, therefore is appears to be the latter case.

Interestingly, the slackening of the precipitation cycle shown Figure 5B roughly coincides with a shift in the annual precipitation cycle. However, changes to the soil moisture cycle appear more clear than those to the precipitation cycle, and this information could be related to the timing of the observed climate transition as determined by clustering, and it could be used to understand whether the shift was one primarily of one or two environmental parameters. This would be especially useful for examining land surface cells that have strong seasonal precipitation events.

Spectral analysis techniques can be a useful means to rapidly identify variant periodicities in geophysical time series, and to diagnose the behavior of model simulations. The periodicity captured by model output, and especially the change in duration and intensity of cycles, is an important feature that can be rapidly analyzed using the methods outlined in this paper.

Bibliography

[1] Hoffman, F.M., Hargrove, W.W., Erickson, D.J.: Using clustered climate regimes to analyze and compare predictions from fully coupled General Circulation Models. Earth Interactions 9(10), 1 (2005)
[2] Washington, W.M., Weatherly, J.W., Meehl, G.A., Semtner Jr., A.J., Bettge, T.W., Craig, A.P., Strand Jr., W.G., Arblaster, J., Wayland, V.B., James, R., Zhang, Y.: Parallel climate model (PCM) control and transient simulations. Climate Dynamics 16, 755–774 (2000)
[3] Hoffman, F.M., Hargrove, W.W.: Multivariate geographic clustering using a Beowulf-style parallel computer. In: Arabnia, H.R. (ed.) Proc. International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 1999), Las Vegas, Nevada, vol. III, pp. 1292–1298. CSREA Press (June 1999)
[4] Hartigan, J.A.: Clustering Algorithms. John Wiley & Sons, New York (1975)
[5] Cadzow, J.A.: Discrete–Time Systems: An Introduction with Interdisciplinary Applications. Prentice-Hall, Inc., Englewood Cliffs (1973)
[6] Torrence, C., Compo, G.P.: A practical guide to wavelet analysis. Bulletin of the American Meteorological Society 79, 61–78 (1998)
[7] Kaiser, G.: A Friendly Guide to Wavelets. Birkhäuser, Boston (1994)
[8] De Moortel, I., Munday, S.A., Hood, A.W.: Wavelet analysis: the effect of varying basic wavelet parameters. Solar Physics 222, 203–228 (2004)
[9] Farge, M.: Wavelet transforms and their applications to turbulence. Annual Review of Fluid Mechanics 24, 395–457 (1992)
[10] Cornette, J.L.: Gauss-vaníček and fourier transform spectral analyses of marine diversity. Computing in Science and Engineering 9(4), 61–63 (2007)
[11] Rohde, R.A., Muller, R.A.: Cycles in fossil diversity. Nature 434, 208–210 (2005)

Seismic Wave Field Modeling with Graphics Processing Units

Tomasz Danek

Department of Geoinformatics and Applied Computer Science,
Faculty of Geology, Geophysics and Environmental Protection
AGH University of Science and Technology,
Al. Mickiewicza 30, 30-059 Krakow, Poland
tdanek@agh.edu.pl

Abstract. GPGPU - general-purpose computing on graphics processing units is a very effective and inexpensive way of dealing with time consuming computations. In some cases even a low end GPU can be a dozens of times faster than a modern CPUs. Utilization of GPGPU technology can make a typical desktop computer powerful enough to perform necessary computations in a fast, effective and inexpensive way. Seismic wave field modeling is one of the problems of this kind. Some times one modeled common shot-point gather or one wave field snapshot can reveal the nature of an analyzed wave phenomenon. On the other hand these kinds of modelings are often a part of complex and extremely time consuming methods with almost unlimited needs of computational resources. This is always a problem for academic centers, especially now when times of generous support from oil and gas companies have ended.

1 Introduction

Recent rapid development of computer based entertainment made graphics cards one of the most important part of whole computer systems. Typical modern personal computer systems and a constantly growing percentage of portable systems are designed to be multimedia centers. Fortunately some of a modern entertainment technology can be used for something more than an entertainment. It is well known that a graphic processing unit is in fact a powerful parallel system dedicated for matrix-to-matrix calculations (rendering). During last years many methods of using this power for calculations were developed. Intensive involvement of two the most important companies at this market: NVIDIA and ATI results in a marvelous dedicated software (e.g. CUDA, Stream SDK) and an astonishing hardware (NVIDIA TESLA). At the same time classic methods based on OpenGL including additions and standard hardware were developed. This paper is focused only on noncommercial, as free as possible, open-source solutions of the later kind. These solutions are usually better for long academic projects because rules of usage are clear and constant which is not always true in the case of commercial software, even if one can use them without any additional costs. One of the typical projects of this kind is a seismic wave field modeling.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 435–442, 2009.

The growing popularity of this method in seismic and seismology (e.g. [1,2]) has been recently connected with the usage of inexpensive HPC clusters (e.g. [3]). But access to supercomputers or HPC clusters is usually limited. It is very common that small local problems, which can be easily solved with parallel computations have to be queued for hours because there are not enough free cluster resources. Moreover, many new extremely computationally expensive methods which use wave field modelings are being developed now. For example full wave form inversion through Monte Carlo sampling sometimes requires hundreds of thousands of models to be computed [2].

2 Seismic Wave Field Modeling

Wave field modeling is an important tool for seismic exploration and seismology. It can be used during all stages of seismic investigations and for various earthquake related analysis. First attempts of using a seismic wave field modeling were undertaken in the seventies by Alford, Kelly and others [4,5]. These attempts were limited to very small models due to the limitations of computers at that time. Even now serial computations for models of a standard exploration scale could last many days. Fortunately wave field modeling is a problem which is easy and effective to solve with parallel systems. In this paper a simple acoustic wave equation was chosen for a test computation:

$$\frac{\partial^2 p}{\partial t^2} - c^2 \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial z^2} \right) = f(x, z, t) \tag{1}$$

where $p(x, z)$ is pressure, $c(x, z)$ is velocity of acoustic wave, t is time and f denotes function which describes pressure change in source.

The second order finite difference approximation of the above equation (without source term) can be written as follows[4]:

$$\begin{aligned} &p(i, j, k+1) = \\ &2(1 - 2\gamma^2)p(i, j, k) - p(i, j, k-1) \\ &+\gamma^2(p(i+1, j, k) + p(i-1, j, k) + p(i, j+1, k) + p(i, j-1, k)) \end{aligned} \tag{2}$$

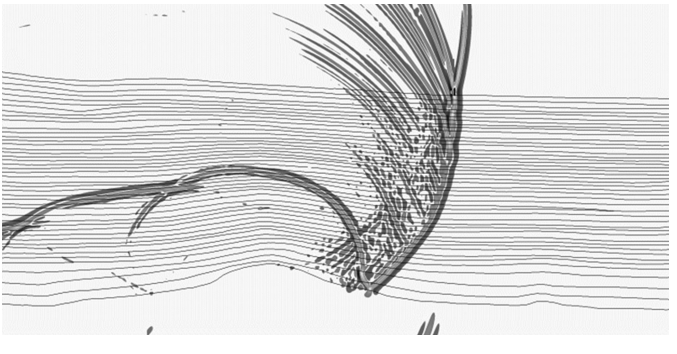


Fig. 1. Example result of acoustic wave field modeling

where $\gamma = c\Delta t/\delta h$, Δt is the time sampling interval, Δh is the distance between the grid points in x and z directions. The stability criterion for the above scheme is: $\gamma \leq 1/\sqrt{2}$.

It is also necessary to add proper border conditions. In this tests typical non-reflective boundary conditions by Reynolds [6] were used.

One of possible modeling result presentations, a snapshot of wave field propagating through complicated geological medium is presented in Figure 1.

3 GPGPU

GPGPU - General-Purpose computation on GPUs is one of the most rapidly developing part of modern computer science. Originally GPUs were created for very specific operations connected only with graphics. Later, when new advanced features - like possibility of floating point parallel operations - were introduced, it became clear that these units can be used for very fast and extremely inexpensive calculations. At the beginning this kind of utilization of GPUs seemed to be only marginal part of the IT market. But now, when the first GPU-based heterogeneous cluster joined the "Top 500", (November 17th, 2008) GPGPU is a mainstream technology with a sophisticated dedicated hardware and an effective commercial software. At the same time many free and open software solutions were created. They made a typical hardware to be much more programmer-friendly (e.g. BrookGPU).

The main source of computational power of a GPU is its design. Most of the transistors are dedicated to a data processing when the caching and data control functionalities are limited [7] (see Figure 2). This high specialization makes a GPU powerful tool for computations but at the same time makes it very weak for many other tasks which are hard to parallelize, like all kind of applications dominated by memory communication [8]. There are also other limitations like amount of memory and texture sizes. Usually all computational problems have to fit into 1GB of memory and 4096 to 8192 texture size.

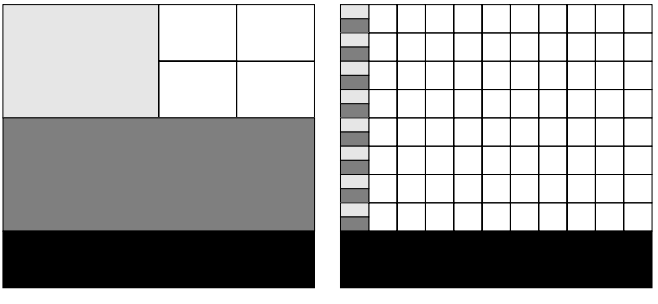


Fig. 2. Comparison of the design differences between CPU (left) and GPU (right) architecture [7]. (white: ALU; light gray: data control; dark gray: cache; black: DRAM).

4 Implementation

The proposed solution was tested on one of the departmental PC cluster, made of 30 computers with low end AGP Nvidia adapters. The best ones were 7300 GT with 256 MB memory. Additional tests were done on PCI Express Nvidia 8600 GT card with 512 MB memory. Other important card parameters are shown in table 1. All cluster machines were working under Linux operating system but all tests were repeated for MS Windows XP for comparison purposes. In both cases the newest possible standard drivers were used. All test codes have been written in C (gcc for Linux and VC++ for Windows) and OpenGL shading language (Listing 1). Data were stored in GL_TEXTURE_RECTANGLE_ARB texture targets which are the most natural targets for computational purposes. It is because of a lack of coordinates normalization and an arbitrary dimensions. In all cases only one floating point number per texel was stored, therefore GL_LUMINANCE texture format and GL_FLOAT_R32_NV internal format were used. For intra-cluster communication mpich implementation of MPI was used. All test runs were done for a simple two layer geological model of various sizes. Wave velocity in upper layer was 1000 m/s and 2000 m/s in lower. Wave motion was modeled for 0.3 second. Distance between grid points was 1 meter

Table 1. Parameters of GPUs used in tests

Model	Chip	Core clock	Memory clock	Memory type	Pixel shader	Vertex shader
7300 GT	G73	400 MHz	1000 MHz	GDDR3 (128Bit)	8	4
8600 GTS	G84	675 MHz	2000 MHz	GDDR3 (128Bit)	up to 32	up to 32

Listing 1. OpenGL shading language implementation of equation 2. vec2 type was used for code clarity. Vp - wave velocity; p and pm - wave field at time t and $t-1$; ds - distance between grid points; dtr - time step.

```
v1=vec2(gl_TexCoord[0].x,gl_TexCoord[0].y);

w=2.0*texture2DRect(p,vec2(v1.x,v1.y)).x
  -texture2DRect(pm,vec2(v1.x,v1.y)).x
  +((dtr*dtr)/(ds*ds))
  *texture2DRect(Vp,vec2(v1.x,v1.y)).x
  *texture2DRect(Vp,vec2(v1.x,v1.y)).x*(
    texture2DRect(p,vec2(v1.x+1.0,v1.y)).x
    +texture2DRect(p,vec2(v1.x-1.0,v1.y)).x
    +texture2DRect(p,vec2(v1.x,v1.y+1.0)).x
    +texture2DRect(p,vec2(v1.x,v1.y-1.0)).x
    -4.0*texture2DRect(p,vec2(v1.x,v1.y)).x
  );
```

which means that 1200 iterations were needed for final result. In all numerical experiments non-reflective boundary conditions were used. In case of GPU computations these kind of conditions are applied by proper `if` statements which makes computations up to 10% slower. Alternatively attenuating boundary conditions can be used but they require an additional texture to be created, which is memory consuming. Additionally all models were calculated conventionally using typical cluster node with Intel Pentium 4 3.0 Ghz processor and 2 GB of memory. In this part of the experiment GNU and Intel compilers were used. Obtained times were a base for unoptimized and optimized speedup calculations.

5 Selected Results and Discussion

In the experiment computational times needed by both of the analyzed GPUs were calculated for Linux and Windows operating systems. The relation between time of computations and model size is shown in Figure 3. The results show that in 8600 GTS card case computations for the very small models are marginally faster under Linux system but for larger models Windows driver is faster. The interesting phenomenon is a rapid increase of the time of computations for the largest analyzed model under Linux. Exactly the same was observed for one of the older Windows drivers, so probably this problem will be fixed in the future. Surprisingly in the case of 7300 GT relation between results is reversed and computations under Linux are usually faster. Figures 4, 5 and 6 show how much faster GPU computations are in comparison with unoptimized and optimized codes run on Intel P4 3.0 GHz CPU . It is clearly visible that for usual scale models GPU computations are 10 (7300 GT) to 50 (8600 GTS) times faster than gcc compiled CPU code with default set of flags (no optimization)(Figure 4). Results of similar experiment but for optimized gcc code are presented in Figure 5. The fastest code was obtained for singe -O3 flag. Executables compiled with more sophisticated options were slower or their average processor time was almost exactly the same. In this case GPU version was up to 20 times faster. The last comparison was done for Intel C compiler (Figure 6). As it was expected icc codes with standard set of flags and autoparallelization were better than those generated with gcc, but still 8600 GTS card was over a dozen times faster. It is important to emphasize that both cards are inexpensive and one can get the cheaper one for much less then 100 USD. The other important phenomenon to discuss is a rapid fall of the acceleration curve for larger model in the case of the 8600 GTS card. GPGPU calculations speedup curves usually have rather unstable shape which is strongly connected with the version of the card driver. It is also important to realize that analyzed case is special and very well suited for GPGPU and one should not expect similar speedups for other kinds of computations.

6 Summary

Possibility of GPGPU computations application in seismic wave field modeling was presented in this paper. It is clearly visible that graphics processing units

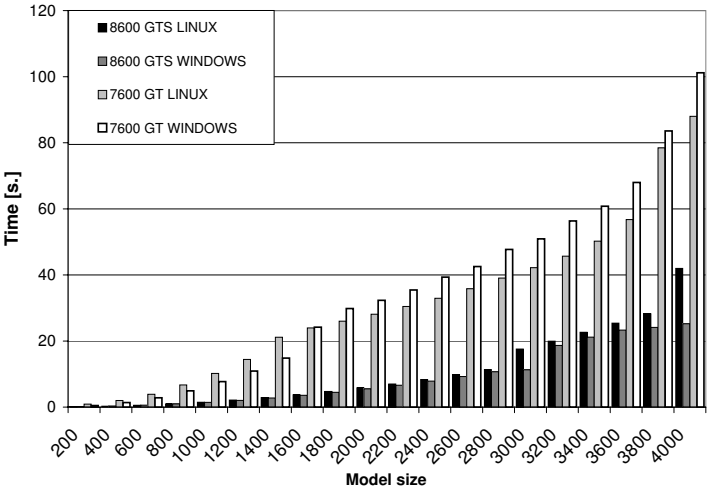


Fig. 3. Relation between time of computations and size of the model for selected hardware and systems. Values on bottom axis represent grid points per side of rectangular model. Black bars: 8600 GTS - Linux; dark gray bars: 8600 GTS - Windows; light gray bars: 7300 GT - Linux; white bars: 7300 GT - Windows.

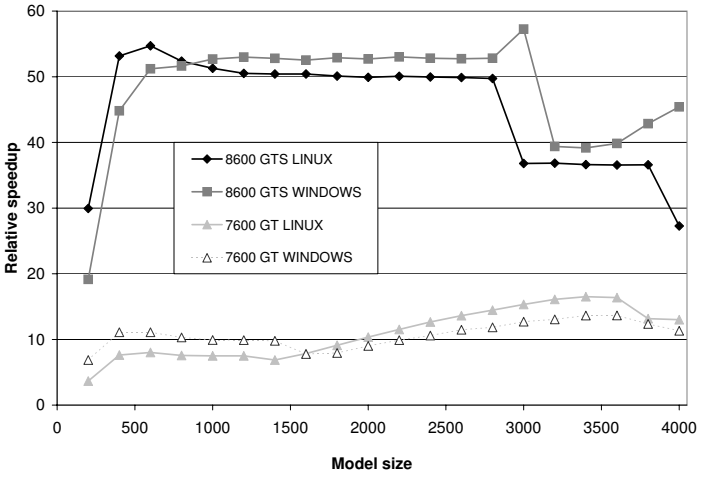


Fig. 4. Relative speedups for selected hardware and systems. The base for speedup calculations is time of conventional CPU computations performed on Intel Pentium 4 3.0 GHz processor - results for unoptimized gcc. Black rectangles: 8600 GTS - Linux; dark gray rectangles: 8600 GTS - Windows; light gray triangles: 7300 GT - Linux; white triangles: 7300 GT - Windows.

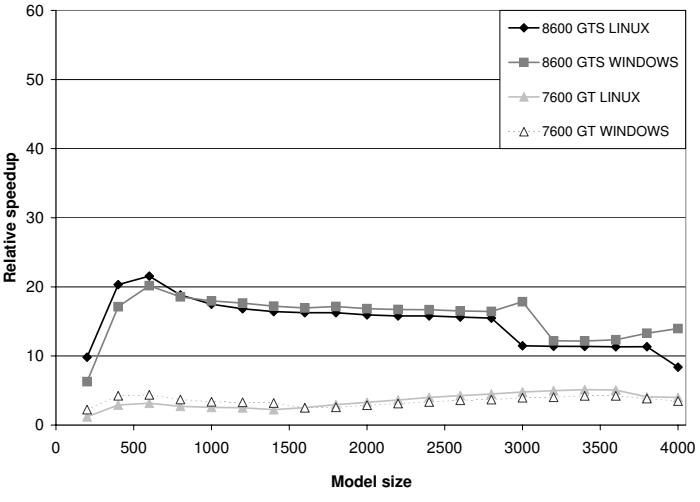


Fig. 5. Relative speedups for selected hardware and systems. The base for speedup calculations is time of conventional CPU computations performed on Intel Pentium 4 3.0 GHz processor - results for the best set of gcc optimization options. Black rectangles: 8600 GTS - Linux; dark gray rectangles: 8600 GTS - Windows; light gray triangles: 7300 GT - Linux; white triangles: 7300 GT - Windows.

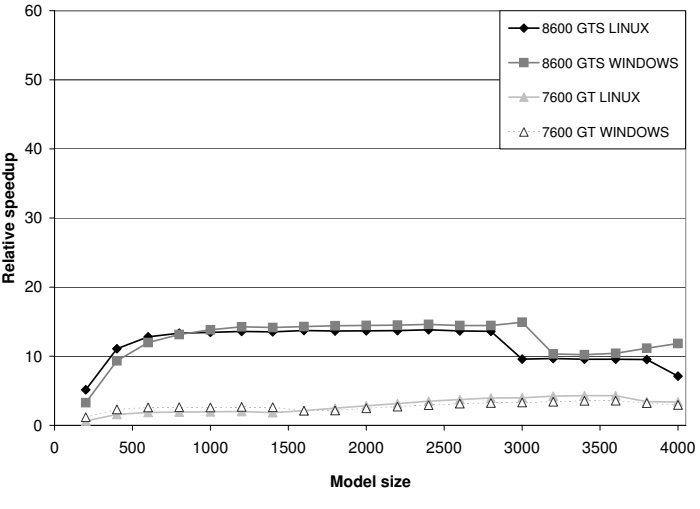


Fig. 6. Relative speedups for selected hardware and systems. The base for speedup calculations is time of conventional CPU computations performed on Intel Pentium 4 3.0 GHz processor - results for the best set of Intel compiler (icc) optimization options. Black rectangles: 8600 GTS - Linux; dark gray rectangles: 8600 GTS - Windows; light gray triangles: 7300 GT - Linux; white triangles: 7300 GT - Windows.

are very effective in this kind of computations. It should be stressed that presented tests are preliminary and other more complicated cases must be studied. Anyway the obtained results are promising and can be a contribution to studies concerning efficiency of alternative modern HPC methods in seismic and seismology.

Acknowledgments. This work was financed by the AGH - University of Science and Technology, Faculty of Geology, Geophysics and Environmental Protection as a part of statutory project number 11.11.140.561.

References

1. Pietsch, K., Marzec, P., Kobylarski, M., Danek, T., Lesniak, A., Tatarata, A., Gruszczyk, E.: Identification of seismic anomalies caused by gas saturation on the basis of theoretical P and PS wavefield in the Carpathian Foredeep, SE Poland. Acta Geophysica 55(2) (2007)
2. Debski, W., Danek, T., Pieta, A., Lesniak, A.: Waveform inversion through the Monte Carlo sampling. In: 31st General assembly of the European Seismological Commission, Hersonissos, Crete (2008)
3. Lesniak, A., Danek, T.: Efficiency of Linux clusters in multi-component elastic wave field modeling in anisotropic media. In: 68th EAGE conference and exhibition, Wien (2006)
4. Alford, R.M., Kelly, K.R., Boore, D.M.: Accuracy of finite - difference modeling of acoustic wave propagation. Geophysics 39(6) (1974)
5. Kelly, K.R., Ward, R.W., Treitel, S., Kelly, K.R., Alford, R.M.: Synthetic seismograms: A finite-difference approach. Geophysics 41(6) (1975)
6. Reynolds, A.C.: Boundary conditions for the numerical solution of wave propagation problems. Geophysics 43 (1978)
7. NVIDIA CUDA Compute Unified Device Architecture Programming Guide. Version 2.0 - 06. 07 (2008)
8. Owens, J.D., Luebke, D., Govindaraju, N., Harris, M., Kruger, J., Lefohn, A.E., Purcell, T.J.: A Survey of General-Purpose Computation on Graphics Hardware. Computer Graphics forum 26(1) (2007)

Dynamic Data Driven Applications Systems – DDDAS 2009

Craig C. Douglas

University of Wyoming, Laramie, WY 82071, USA
Yale University, New Haven, CT 06520-8285, USA

Abstract. This workshop covers several aspects of the Dynamic Data Driven Applications Systems (DDDAS) concept, which is an established approach defining a symbiotic relation between an application and sensor based measurement systems. Applications can accept and respond dynamically to new data injected into the executing application. In addition, applications can dynamically control the measurement processes. The synergistic feedback control-loop between an application simulation and its measurements opens new capabilities in simulations, e.g., the creation of applications with new and enhanced analysis and prediction capabilities, greater accuracy, longer simulations between restarts, and enable a new methodology for more efficient and effective measurements. DDDAS transforms the way science and engineering are done with a major impact in the way many functions in our society are conducted, e.g., manufacturing, commerce, transportation, hazard prediction and management, and medicine. The workshop will present such new opportunities as well as the challenges and approaches in technology needed to enable DDDAS capabilities in applications, relevant algorithms, and software systems. The workshop will showcase ongoing research in these aspects with examples from several important application areas.

1 The Scope of the Workshop

More and more applications are migrating to a data-driven paradigm including hazard management, terrorist event handling, contaminant tracking, chemical process plants, petroleum refineries, well bores, and nuclear power plants. In each case sensors produce large quantities of telemetry that are fed into simulations that model key quantities of interest. As data are processed, computational models are adjusted to best agree with known measurements. If properly done, this increases the predictive capability of the simulation system. This allows what-if scenarios to be modeled, disasters to be predicted and avoided with human initiated or automatic responses, and the operation of the plants to be optimized. As this area of computational science grows, a broad spectrum of application areas will reap benefits. Examples include enhanced oil recovery, optimized placement of desalination plants and other water intakes, optimized food production, monitoring the integrity of engineered structures and thus avoiding failures, and real time traffic advice for drivers. These are but a few of countless examples.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 445–446, 2009.

As is the case in other data intensive arenas, visualization plays a key role in DDDAS. Visualization is used at all stages: setting up data and initial and/or boundary conditions, seeing and analyzing results, and steering computations.

Data-driven computational science is ripe for multidisciplinary research to build applications, algorithms, measurement processes, and software components from which tools can be developed to solve diverse problems of regional and international interest. The advances that will result, including enhanced repositories of software components and applications, will be of great value to industry and governments, and will set the stage for further valuable research and development. A comprehensive list of ongoing state of the art projects is kept up to date on http://www.dddas.org in the projects area.

Several research thrusts in which advances should significantly enhance the ability of data-driven computational science to bring its tremendous benefits to a wide array of applications. These research thrusts, which are described below in more detail, are:

- Effective *assimilation* of streams of data into ongoing simulations.
- *Interpretation, analysis, and adaptation* to assist the analyst and to ensure the most accurate simulation.
- *Cyberinfrastructure* to support data-driven simulations.

These three areas interact with two other research fields symbiotically: (1) forward multiscale modeling and simulation, and (2) deterministic and statistical methods in inverse problems.

Research areas (1) and (2) combined with (3) DDDAS must work within the context of uncertainty and will benefit from the development of statistically sound, unified treatments of uncertainties. For example, in forward multiscale modeling and simulation, input data are uncertain and these uncertainties should be propagated to uncertainties in output quantities of interest. In an inverse problem, proper treatment of measurement uncertainties and errors must be integrated with treatment of uncertainties associated with bforwardb models. be treated systematically. In a data-driven application, all of these uncertainties are present and must.

Data management in a DDDASis typically supported by tools for data acquisition, data access, and data dissemination. Data acquisition tools retrieve the real time or near real time data, processing and storing them. Data access tools provide common data manipulation support, e.g., querying, storing, and searching, to upper level models. Data dissemination tools read data from the data store, format them based on requests from data consumers, and deliver formatted data to data consumers.

Software re-use, supporting separation of concerns, and real time quality support for applications will increase reliability, reduce development time, and teach students how to be much more productive.

DDDAS and similar methodologies are *the* paradigm of the truly data rich information age upon us.

Characterizing Dynamic Data Driven Applications Systems (DDDAS) in Terms of a Computational Model

Frederica Darema

National Science Foundation, Arlington, VA USA
darema@nsf.gov

Abstract. The DDDAS (Dynamic Data Driven Applications Systems) concept creates new capabilities in applications and measurements, through a new computational paradigm where application simulations can dynamically incorporate and respond to online field-data and measurements, and/or control such measurements. Such capabilities entail dynamic integration of the computational and measurement aspects of an application in a dynamic feed-back-loop, leading to unified SuperGrids of the computational and the instrumentation platforms. Examples of advances in applications capabilities enabled through DDDAS over traditional computational modeling methods, and advances in measurements methods, and instrumentation and sensor network systems, have appeared extensively in the literature. This paper concentrates in discussing a computational model representing the unified DDDAS computation-measurement environments, and asymptotic cases leading to traditional computational environments, data assimilation, and traditional control systems.

Keywords: computational model, applications, measurements, dynamic runtime, sensors, grids.

References

1. NSF Workshop (March 2000), http://www.cise.nsf.gov/dddas
2. Douglas, C.C.: (2000-2009), http://www.dddas.org
3. Darema, F.: Grid Computing and Beyond: The Context of Dynamic Data Driven Applications Systems. In: Proceedings of the IEEE, Special Issue on Grid Computing (March 2005)
4. Darema, F.: Dynamic Data Driven Applications Systems: A New Paradigm for Application Simulations and Measurements. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3038, pp. 662–669. Springer, Heidelberg (2004)
5. Darema, F.: Dynamic Data Driven Applications Systems: New Capabilities for Application Simulations and Measurements. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3515, pp. 610–615. Springer, Heidelberg (2005)
6. Darema, F.: Introduction to the ICCS 2006 Workshop on Dynamic Data Driven Applications Systems (2006)

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 447–448, 2009.

7. NSF Sponsored Workshop on DDDAS-Dynamic Data Driven Applications Systems, January 19-20 (2006), http://www.cise.nsf.gov/dddas
8. DDDAS-Dynamic Data Driven Applications Systems Program Solicitation, NSF 05-570, http://www.cise.nsf.gov/dddas
9. Darema, F.: Novel Drivers for Future InterNets: Dynamic Data Driven Applications Systems (DDDAS). In: 1st IEEE Workshop on Enabling Future Service-Oriented Internets GLOBECOM (2007)
10. Darema, F.: DDDAS Computational Model and Environments. Journal of Algorithms and Computational Technology (2009/2010) (to appear)

Enabling End-to-End Data-Driven Sensor-Based Scientific and Engineering Applications*

Nanyan Jiang and Manish Parashar

Center for Autonomic Computing (CAC) &
The Applied Software Systems Laboratory (TASSL)
Department of Electrical and Computer Engineering
Rutgers University, Piscataway NJ 08855, USA
{nanyanj,parashar}@rutgers.edu

Abstract. Technical advances are leading to a pervasive computational infrastructure that integrates computational processes with embedded sensors and actuators, and giving rise to a new paradigm for monitoring, understanding, and managing natural and engineered systems – one that is information/data-driven. However, developing and deploying these applications remains a challenge, primarily due to the lack of programming and runtime support. This paper addresses these challenges and presents a programming system for end-to-end sensor/actuator-based scientific and engineering applications. Specifically, the programming system provides semantically meaningful abstractions and runtime mechanisms for integrating sensor systems with computational models for scientific processes, and for in-network data processing such as aggregation, adaptive interpolation and assimilations. The overall architecture of the programming system and the design of its key components, as well as its prototype implementation are described. An end-to-end dynamic data-driven oil reservoir application that combines reservoir simulation models with sensors/actuators in an instrumented oilfield is used as a case study to demonstrate the operation of the programming system, as well as to experimentally demonstrate its effectiveness and performance.

1 Introduction

Technical advances are rapidly leading to a revolution in the type and level of instrumentation of natural and engineered systems, and are resulting in a pervasive computation ecosystem that integrates computers, networks, data archives, instruments, observatories, experiments, and embedded sensors and actuators. This in turn is enabling a new paradigm for monitoring, understanding, and managing natural and engineered systems – one that is information/data-driven and

* The research presented in this paper is supported in part by National Science Foundation via grants numbers CNS 0723594, IIP 0758566, IIP 0733988, CNS 0305495, CNS 0426354, IIS 0430826 and ANI 0335244, and by Department of Energy via the grant number DE-FG02-06ER54857, and was conducted as part of the NSF Center for Autonomic Computing at Rutgers University.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 449–459, 2009.

that symbiotically and opportunistically combines computations, experiments, observations, and real-time information to model, manage, control, adapt, and optimize.

Several application domains, such as waste management [1], underwater ocean phenomenon monitoring [2], city-wide structural monitoring [3] and end-to-end soil monitoring system [4], are already experiencing this revolution in instrumentation, and can potentially allow new quantitative synthesis and hypothesis testing in near real time as data streams in from distributed instruments. However, these application present many new and challenging requirements due to (1) the data volume and rates, (2) the uncertainty in this data and the need to characterize and manage this uncertainty and (3) the need to assimilate and transport required data (often from remote sites over low bandwidth wide area networks) in near real-time so that it can be effectively integrated with (running) computational models and analysis systems. A key challenge is the lack of effective programming and system software supports that can support these applications in an end-to-end manner. As a result, in most existing instrumented systems data acquisition is a separate offline process and this data is typically used in a post-processing manner [5].

The overall goal of the research effort presented in this paper is to investigate sensor system middleware and programming support that will enable distributed networks of sensors to function, not only as passive measurement devices, but as intelligent data processing instruments, capable of data quality assurance, statistical synthesis and hypotheses testing as they stream data from the physical environment to the computational world [5]. Further, application should be able to interact with the sensor system to control sensing and data processing behaviors. The programming systems enables sensor-driven applications at two levels. First, it provides programming abstractions for integrating sensor systems with computational models for scientific processes (e.g. biophysical, geophysical processes) and with other application components in an end-to-end experiment. Second, it supports programming models and systems for developing in-network data processing mechanisms. The former supports complex querying of the sensor system, while the latter enables development of in-network data processing mechanisms such as aggregation, adaptive interpolation and assimilations, both via semantically meaningful abstractions. The research is driven by the management and control of subsurface geosystems, such as managing subsurface contaminants at the Ruby Gulch waste repository [1] and management and optimization of oil reservoirs [6]. Crosscutting requirements of these applications include multi-scale, multi-resolution data access, data quality and uncertainty estimation, and predictable temporal response to varying application characteristics.

The focus of this paper is on the end-to-end abstractions provided by the programming system, and on how they can be used to enable scientific/engineering applications to discover, query, interact with, and control instrumented physical systems in a semantically meaningful way. Specifically, this papers describes the design and operation of the *GridMap* abstraction, and demonstrates its usage

and effectiveness using an *Instrumented Oilfield* application as a case study. A prototype programming system has been implemented. An experimental evaluation using this prototype is also presented.

The rest of the paper is organized as follows. Section 2 gives an overview of the overall programming system and the *GridMap/iZone* abstractions it provides. Section 3 presents instrumented oilfield application case study. Section 4 presents an experimental evaluation of the programming system and the abstraction provided. Section 5 discusses related work. Section 6 concludes the paper and outlines current and future work.

2 The *GridMap* and *iZone* Programming Abstractions

Scientific applications often require measurements at pre-defined grid points, which are often different from the locations of the raw data provided directly by the sensor network. As a result, the sensor-driven scientific and engineering applications require a virtual layer, where the logical representation of the state of the environment provided to the applications may be different from the physical representation of raw measurements from sensor network. The abstractions described in this section enable applications to specify such a virtual layer and the models (e.g. regression models, interpolation functions, etc.) that should be used to estimate data on the virtual layer from sensor readings, as well to develop in-network implementations of the data estimation mechanisms.

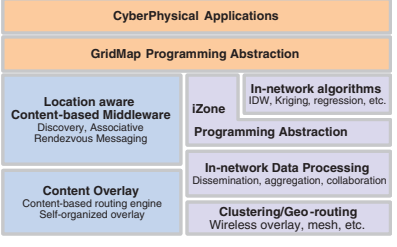


Fig. 1. A schematic overview of the programming system

A schematic overview of the overall programming system architecture is presented in Fig. 1. It consists of a two-level programming abstraction, the end-to-end *GridMap* programming abstraction that enables computational applications to access and integrate sensor data into their models, and the in-network *iZone* programming abstraction to enable the development of scalable in-network data processing mechanisms. The underlying middleware provides an in-network data processing engine, which supports efficient data dissemination, aggregation and collaboration in dynamics, resource-constraint heterogeneous sensor networks.

The *GridMap* Abstraction. The *GridMap* abstraction consists of two operators. The first operator allows the application to construct a virtual grid (a *GridMap*), corresponding to the computational grid used by the computational models, on the instrumented domain. Once this virtual grid has been overlayed on the sensor system, the application can use the second operator to query data corresponding to a region of interest on this virtual grid. The interface of this second operator includes a specification of the method (e.g., interpolator) that should be used to estimate data at a grid point in the region of interest using physical data from sensors that are in the neighborhood of the point. The operator also includes parameters such as the size of neighborhood that should be used in the estimation, and what are the constraints on the accuracy and cost of the estimation.

The *GridMap* operators include end-to-end query operations, i.e., *query*, *notify*, *retrieve* as well as operators to construct, modify and delete the *GridMap* , i.e., *init*, *delete*, *refine* and *coarsen*. The parameters of the *query* operator include a specification of the region of interest within the *GridMap* and the interpolation function that should be used to compute values at the grid points of interest form the sensor values. The execution of this operator results in a query message being routed to relevant nodes (i.e., cluster heads) in the sensor network. The query specification is then matched against existing profiles, and if required, appropriate in-network operators are invoked. The *notify* operator is used to register notification requests, for example, and application may be interested in be notified if the maximum sensor reading in a region of the *GridMap* exceeds a certain threshold. The application can retrieve previously queried values using the *retrieve* operator. The *init* operator is used to initialize the grid points associated with *GridMap*. The *refine* operator modifies an existing *GridMap* by adding more grid points to increase the resolution of the *GridMap*. The *coarsen* operator modifies an existing *GridMap* by suspending some of the virtual grid points to effectively reduce the resolution of the *GridMap*. Note that, both *refine* and *coarsen* operators do not re-construct of the entire *GridMap* which makes them more efficient in dynamically changing physical environments.

The *iZone* Abstraction. The *iZone* abstraction in turn, enables the implementation of the estimation functions. The *iZone* itself is a representation of the neighborhood that is used to compute a grid point, and may be specified using a range of coordinates, a function, etc. The *iZone* abstraction also provides operators, such as *discover*, *expand*, and *shrink* for obtaining sensors corresponding to the region of interest as well as for defining in-network processing operators, such as *get*, *put*, and *aggregate*, to compute a desired grid point from sensor values from this region as summarized in [7].

Note that, with *GridMap/iZone* programming system, user-defined function can be implemented with *iZone* operators, which can then be applied in a straightforward fashion as a function operator on the actual running environment with *GridMap* operators.

3 An End-to-End Oil Reservoir Application Using *GridMap/iZone* Abstractions

In this section, we demonstrate how the *GridMap/iZone* abstractions can be used to implement an end-to-end oil reservoir application that combines reservoir simulation models with sensors/actuators in an instrumented oilfield.

Subsurface behavioral surveillance and sensing is becoming available in an increasing number of environmental and energy reservoir applications. The deployment of sensors is offering unlimited possibilities to monitor and obtain a dynamic understanding of the different processes taking place at different spatial and temporal scales. The proposed programming abstractions and systems software solutions can enable the end-to-end management process for detecting and tracking reservoir changes, assimilating and inverting data for determining reservoir properties, and providing feedback to enhance temporal and spatial resolutions and track other specific processes in the subsurface, so as to ensure new optimal operation (in terms of profitability, safety or environmental impact).

An overview of this application scenario is shown in Fig. 2. The figure illustrates the steps involved in constructing a closed control loop for optimal reservoir management, including computational processes for issuing queries to instrumented oil reservoir, retrieving the relevant data and integrating it with the simulation processes, and making appropriate decisions for updating oil production policy.

The evolution of pressure and concentration in the oil field during production are simulated using comprehensive mathematical models of the subsurface. As the simulations evolve, these models periodically update pressure and concentration distributions in particular regions (e.g., regions requiring adaptive refinement due to high errors) from the oil field. Sensors deployed in the oil field monitor and retrieve current pressures and/or concentrations in regions of interest. The results of the simulations are then used by the production optimization process to generate optimal configuration (i.e. production rate, gas, water pressures, etc.). The realization of these steps using the *GridMap/iZone* abstractions are briefly described below.

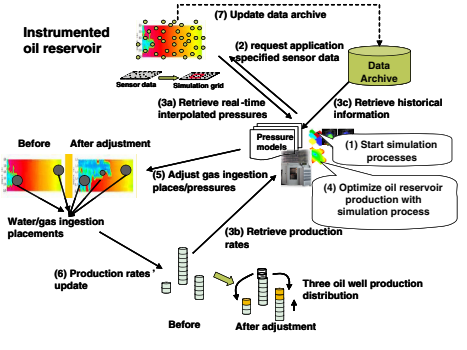


Fig. 2. Overview of an end-to-end oil reservoir application

The application first uses the *GridMap* "init" operator to construct a virtual sensor grid overlayed on the oilfield, to match the grid used by the simulations models. For example, a *GridMap* instance constructed using the specification $<x_{lb}{:}\delta_x{:}x_{ub},\ y_{lb}{:}\delta_y{:}y_{ub}>$ represents a two dimensional rectangular virtual grid with its left bottom corner at (x_{lb}, y_{lb}) and top right corner at (x_{ub}, y_{ub}), and with a spacing of δ_x and δ_y along the x and y dimension respectively. The application can also specify the interpolation method, for example, "closest neighbor", "IDW" or "Kriging", to be used to compute data values at the virtual grid points using sensor values.

The application can now "query" the sensor system using regions of the virtual grid. The application query specifies the data types of interest (e.g., pressure, concentration), error thresholds, etc., using the syntax described in Section 2. The queries are forwarded by the runtime system to appropriate sensors nodes. For each virtual grid point, *iZones* are constructed by discovering all relevant sensors. The data retrieved from these sensors is then interpolated onto the required virtual grid points using the specified interpolation method and the in-networks mechanisms provide by an *iZone*. The resulting data values on the virtual grid are then returned to the application.

The application can now continue the simulations using the updated data, and along with current production rate, historical data, etc., to, for example, predict changes in oil reservoir, evaluate different configurations, and optimize desired objective functions such as maximizing production rate and/or minimizing cost (e.g. gas ingestion cost).

4 Implementation and Experimental Evaluation

4.1 Implementation Overview

The current prototype implementation of *GridMap/iZone* programming system consists of two key parts. The sensor network component is implemented using the 802.11 protocol and standard location based clustering to construct a two level self-organizing overlay of sensor. This component implements the mechanisms for sensor discovery, query dissemination, data gathering and aggregation and in-network data processing. It has been prototyped using sensors emulated on the Orbit wireless testbed [8]. The wide-area component is built using Java and on top of the JXTA peer-to-peer substrate [9] and deployed on the Rutgers campus Grid. This component integrates computations processes (i.e. simulations), data archives and user subsystem to the sensor system through gateway nodes. Queries issued by the computational process are routed to the appropriate sensor nodes (and aggregated and interpolated data values routed) via the gateway and cluster heads. The rest of this section focuses on an experimental evaluation using end-to-end application scenarios.

4.2 Experimental Evaluation

The parameter estimation and oil reservoir optimization scenarios in an instrumented oilfield (described in Section 3) is implemented using the prototype of

the *GridMap/iZone* programming system described above and is used for the experiments presented in this section. The objectives of the experiments are not only to demonstrate the ability of *GridMap/iZone* to support the integration of computational processes with run-time in-network processing of sensor data, but also to evaluate the effectiveness and efficiency of the prototype implementation for realistic scenarios. The scenarios in the experiments are driven by a real-world sensor dataset obtained from an instrumented oil field with 2000 sensors, and consisting of pressure measurements sampled 96 time per day. A two-tiered sensor overlay with 40 clusters was emulated on 40 nodes of the Orbit wireless testbed so that each cluster ran on a single node of the testbed. The sensors are assumed to be randomly distributed in the sensors field. The computational processes ran on compute cluster located at a different campus at Rutgers.

The end-to-end cost of an interaction between a computational process and the sensor system consists of 5 parts: (1) the cost of issuing the *GridMap* query by a computational process, and routing it through the network to the sensor network gateway; (2) the cost of forwarding the query from the gateway to all relevant sensors associated with the *GridMap*; (3) cost of in-sensor-network computations (e.g., interpolation) associated with a query; (4) the cost of aggregating the data and forwarding it to the gateway; and (5) the cost returning the results back to the computational process. The experimental evaluation in this paper focuses on the in-sensor-network costs, i.e., (2), (3) and (4). These costs are measured in terms of the number of messages, the number of hops per message and the volume of data transferred, and the impact of overall size of the *GridMap*. The tradeoffs between the accuracy of data estimation and associated costs are also evaluated.

Communication Costs. This experiment measures the communication costs, in terms of average number of hops, of querying all the sensors associated with a *GridMap*. The first set of experiments keep the size of the *GridMap* fixed at 40ft x 175ft, and varies the radio range of the sensors. The plots in Fig. 3(a) show that, as the radio range increases, the average number of hops decrease. The next set of experiments measured the impact of increasing the size of the *GridMap* on the average number of messages between clusters in the sensor network for two different radio ranges. The plots in Fig. 3(b) shows that, as the size of the

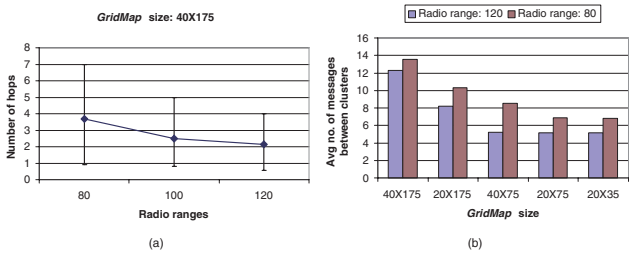


Fig. 3. Communication costs of querying all the sensors associated with a *GridMap*

GridMap increases, so does the average number of messages required per update. For example, when the size of *GridMap* is doubled,e.g., from 20ft x 175ft to 40ft x 175ft in the experiment, the average number of messages increase by about 1.5 times. Additionally, as the size of the *GridMap* becomes smaller (e.g. 20ft x 75ft and 20ft x 35ft), the number of messages does not change much, partially because almost all of the involved clusters of the given *GridMap* are within a single hop of each other.

Tradeoff between Accuracy and Communication Costs. In this experiment, we examine the tradeoff between accuracy and communication costs for cases where the data processing (i.e., interpolation) is done within the sensors system and external to the sensor system (for example, at a remote server). The latter case represents the conventional approach where raw data is collected from the sensor network and transferred to a server where required processing is performed offline. In this case the costs measured in terms of the data volume are transferred to the gateway. The accuracy of interpolation is determined in terms of the interpolation error and depends on the interpolation mechanism used. To ensure a fair comparison, the same data processing is used in both cases.

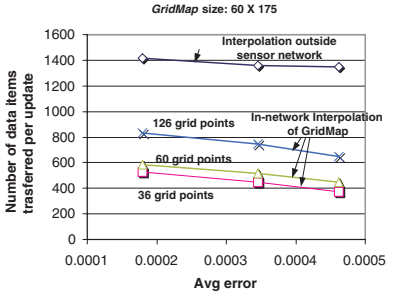


Fig. 4. Tradeoff between accuracy and communication costs

As seen in Fig. 4, increasing the required accuracy increase the volume of communication. For in-network interpolation, this is because more data is required for each *iZone*. Furthermore, as the size of the *GridMap* increases, the volume of communication also increases. Note that the communication cost does not grow linearly with the number of grid points over the same *GridMap*. This is partially because part of the data is shared between neighboring *iZones*, therefore eliminating the need for some of the communications when computing the grid points.

For the case where the interpolation is performed outside the sensor network, the data volume does not change significantly with increasing accuracy since only a small additional amount of data is needed. A key observation is that for the same level of accuracy, the communication volume is significantly large when the interpolation is performed outside the sensor network, which also implies increased bandwidth consumption, latencies as well as energy costs.

This demonstrate a key benefit of in-network data processing, specially in the case of application requiring near real-time analysis of and reaction to sensed data.

5 Related Work

There has been a significant body of research focused on programming support for sensor networks. Several systems [10,11] provide abstractions for specifying the local behaviors of sensors, e.g., state programming [10], and CLB [12] and abstract region [11]. In the *macroprogramming* approach, global behaviors are specified and the programming system generates the local behaviors and necessary interactions, e.g., Kairos [13] and ATaG [14]. *Database-oriented* approaches provide abstractions that view the sensor network as a virtual database system and provide SQL-like interfaces for querying the networks, e.g., TinyDB [15]. The related *data streaming* approach, supported by TelegraphCQ [16], views data as information data streams, and applications monitor and react to them as they pass through the network. Other related systems include the soil ecology monitoring system [4] and the ring buffer network bus (RBNB) DataTurbine [17], which are data collection or management prototypes that uses wireless sensor systems as the component of an end-to-end system.

The programming systems discussed above have to be extended to support end-to-end sensor-driven applications and the interactions between computational models and the sensor system, and address requirements discussed in Section 1. Ideally, a scientist should only have to specify application data requirements using high-level abstractions, and the system should transform these requirements into appropriate operations, interactions and coordination within the sensor system to transform the sensed data so as to match these requirements. The *GridMap/iZone* is such a programming system to provide semantically meaningful abstractions and runtime mechanisms for integrating sensor systems with computational models for scientific processes by essentially virtualizing the sensor field to match its representation used by the computational model. The in-network data processing supports aggregation, adaptive interpolation and assimilations.

6 Conclusion

This paper presented a programming system for end-to-end dynamic data-driven sensor/actuator-based scientific and engineering applications. Specifically, the programming system provides the end-to-end *GridMap* abstraction that enables computational applications to access and integrate sensor data into their models, and the in-network *iZone* abstraction to enable the development of scalable in-network data processing mechanisms. The paper described the overall architecture of the programming system and the design of its key components, as well as its prototype implementation. An end-to-end oil reservoir application

that combines reservoir simulation models with sensors/actuators in an instrumented oilfield was used as a case study to demonstrate the operation of the programming system, as well as to experimentally demonstrate its effectiveness and performance. We are currently working on using the programming system to enable other sensor-driven applications in other domains.

References

1. Parashar, M., Matossian, V., Klie, H., Thomas, S.G., Wheeler, M.F., Kurc, T., Saltz, J., Versteeg, R.: Towards dynamic data-driven management of the ruby gulch waste repository. In: Proceedings of the Workshop on DDDAS, International Conference on Computational Science, ICCS (2006)
2. Johnson, K.S., Needoba, J.A., Riser, S.C., Showers, W.J.: Chemical sensor networks for the aquatic environment. Chemical Review (2007)
3. Kottapalli, V.A., Kiremidjiana, A.S., Lyncha, J.P., Carryerb, E., Kennyb, T.W., Lawa, K.H., Lei, Y.: Two-tiered wireless sensor network architecture for structural health monitoring. In: SPIE's 10th Annual International Symposium on Smart Structures and Materials (2003)
4. Szlavecz, K., Terzis, A., Musaloiu-E., R., Cogan, J., Small, S., Ozer, S., Burns, R., Gray, J., Szalay, A.S.: Life under your feet: An end-to-end soil ecology sensor network, database, web server, and analysis service. MSR-TR-2006-90 (2006)
5. Jiang, N., Parashar, M.: Programming support for sensor-based scientific applications. In: Proceedings of the Next Generation Software (NGS) Workshop in conjunction with the 22nd IPDPS (2008)
6. Klie, H., Bangerth, W., Gai, X., Wheeler, M.F., Stoffa, P., Sen, M., Parashar, M., Catalyurek, U., Saltz, J., Kurc, T.: Models, methods and middleware for grid-enabled multiphysics oil reservoir management. In: Engineering with Computers. Springer, Heidelberg (2006)
7. Jiang, N., Parashar, M.: In-network data estimation mechanisms for sensor-driven scientific applications. In: Proceedings of the 15th International Conference on High Performance Computing, HiPC (2008)
8. ORBIT: ORBIT testbed. Internet: http://www.orbit-lab.org/
9. JXTA: Project JXTA. Internet: http://www.jxta.org
10. Liu, J., Chu, M., Liu, J., Reich, J., Zhao, F.: State-centric programming for sensor-actuator network systems. IEEE Pervasive Computing 2(4), 50–62 (2003)
11. Welsh, M., Mainland, G.: Programming sensor networks using abstract regions. In: Proceedings of the First USENIX/ACM Symposium on Networked Systems Design and Implementation, NSDI 2004 (2004)
12. Jiang, N., Schmidt, C., Matossian, V., Parashar, M.: Enabling applications in sensor-based pervasive environments. In: Proceedings of the 1st Workshop on Broadband Advanced Sensor Networks, BaseNets (2004)
13. Gummadi, R., Gnawali, O., Govindan, R.: Macro-programming wireless sensor networks using Kairos. In: Prasanna, V.K., Iyengar, S.S., Spirakis, P.G., Welsh, M. (eds.) DCOSS 2005. LNCS, vol. 3560, pp. 126–140. Springer, Heidelberg (2005)
14. Bakshi, A., Prasanna, V.K., Reich, J., Larner, D.: The abstract task graph: A methodology for architecture-independent programming of networked sensor systems. In: Workshop on End-to-End, Sense-and-Respond Systems, Applications, and Services, EESR 2005 (2005)

15. Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W.: TinyDB: an acquisitional query processing system for sensor networks. ACM Transactions on Database System 30(1), 122–173 (2005)
16. Chandrasekaran, S., Cooper, O., Deshpande, A., Fanklin, M.J., Hellerstein, J.M., Hong, W., Krishnamurthy, S., Madden, S., Raman, V., Reiss, F., Shah, M.: TelegraphCQ: Continuous dataflow processing for an uncertain world. In: Proceedings of CIDR Conference (2003)
17. Tilak, S., Hubbard, P., Miller, M., Fountain, T.: The ring buffer network bus (RBNB) dataturbine streaming data middleware for environmental observing systems. In: The 3rd IEEE International Conference on e-Science (2007)

Feature Clustering for Data Steering in Dynamic Data Driven Application Systems

Alec Pawling and Greg Madey

Department of Computer Science and Engineering
University of Notre Dame
Notre Dame, IN, USA, 46556

Abstract. In this paper, we describe how feature clustering on real-world cell-phone data can be used to locate the impact area of emergency events. We first examine the effect of two emergency events on the call activity in the areas surrounding the events. We then investigate how the time series of the affected areas behave relative to the time series of their respective neighboring areas. Finally, we examine the differences in hierarchical clusterings of the time series of the affected areas and neighboring areas.[1]

1 Introduction

The Wireless Phone-based Emergency Response (WIPER) system is a proof-of-concept Dynamic Data Driven Application System (DDDAS) designed to leverage real-time streaming cell phone data to provide high-level information about an emergency situation to emergency response managers. WIPER consists of modules for automatically detecting emergency events and for running and validating predictive simulations of potential outcomes [1,2,3,4]. Schoenharl and Madey [5] describe an approach for on-line simulation validation for WIPER using streaming cell phone data as it becomes available. In this paper we address the problem of identifying the area for which the simulations should be run.

In an emergency situation, it is likely that the area of interest is small relative to the total coverage area of the cell phone network. Running predictive simulations for the entire coverage area is problematic in terms of computational requirements and the amount of data produced that must in turn be validated and presented to emergency response managers. In this paper we describe an approach for identifying the area affected by an emergency using feature clustering. We illustrate the effectiveness of this approach using two case studies of emergency events that appear in real-world cell phone data.

2 Related Work

Dynamic Data Driven Application Systems (DDDAS) are characterized by their ability to incorporate new data into running models and simulations as they

[1] This material is based upon work supported by the National Science Foundation, CISE/CNS-DDDAS, Award #CNS-0540348.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 460–469, 2009.

become available and to steer data collection, enabling the simulations to receive and utilize the most relevant data [6,7]. Plale *et al.* [8] use the amount of variance in an ensemble of weather forecast simulations to collect additional data and direct additional computational resources to the areas where additional simulation runs are needed. Flikkema *et al.* [9] uses data models to filter observations at the sensors. In this case, the interesting observations are those that do not match the data model, and it is these that are transmitted for further processing.

WIPER receives a single data stream of cell phone usage information that contains a time stamp, de-identified values indicating the individuals making and receiving the call, and the tower the caller's phone is communicating with. We have the latitude, longitude, and postal code of each tower, and we link this information with the call data. From this data stream, we generate a set of time series for spatially disjoint areas using the tower location information. For each spatial area, we count the number of calls made in 10 minute intervals, producing a vector of non-negative integers for each time step.

We can view this series of vectors as a data set for machine learning algorithms. Let the data set $\mathbf{D}$ be an $n \times m$ matrix with n data items and m features. We can view the problem of identifying the columns of interest, which corresponds to an area in the real world, as the feature selection problem.

Feature selection is the process of identifying the best subset of available features of a data set for machine learning algorithms. Feature selection serves to improve the quality of machine learning models, reduce the computation required to train and utilize these models, and provide a better understanding of the model. One approach to feature selection is to combine similar features using a clustering algorithm [10]. Feature clustering has been used to reduce large feature spaces for applications such as text mining [11].

Data clustering is an unsupervised machine learning method for grouping the rows of a data set $\mathbf{D}$ based on some distance measure. Hierarchical algorithms identify a nested set of partitions in the data. Most hierarchical methods take an agglomerative approach, meaning that there are initially n clusters, each containing one data item in $\mathbf{D}$. These clusters are iteratively merged until all of the data items belong to the same cluster. Popular agglomerative clustering algorithms include single-link and complete-link. These approaches may be implemented using a graph where the data items are represented as vertices and edges are added between two vertices in increasing order of distance between the two corresponding data items. At each step, the clusters in the single-link approach are the connected components and the clusters in the complete-link approach are the completely connected components [12].

Feature clustering applies clustering techniques to the transpose of a data set. Rodrigues *et al.* [13] describe an algorithm for clustering the features of a data stream. The algorithm is a divisive-agglomerative algorithm that uses a dissimilarity measure based on correlation along with a Hoeffding bound to determine when clusters are split. The algorithm utilizes the fact that the pairwise correlation of the time series, $corr(\boldsymbol{a}, \boldsymbol{b})$, can be computed using a small number

of sufficient statistics. The key observation by Rodrigues *et al.* [13] is that it is only necessary to maintain a small number of values to compute the correlation of two time series. For each time series it is necessary to keep track of $\sum_{i=1}^{n} a_i$, $\sum_{i=1}^{n} b_i$, $\sum_{i=1}^{n} a_i^2$, and $\sum_{i=1}^{n} b_i^2$. For each pair of time series $\sum_{i=1}^{n} a_i b_i$ must be updated with the arrival of each data item. Additionally, the number of data items that have arrived so far, n, must be known. Rodrigues *et al.* [13] use correlation distance, $diss(\boldsymbol{a}, \boldsymbol{b}) = 1 - corr(\boldsymbol{a}, \boldsymbol{b})$, as a dissimilarity measure.

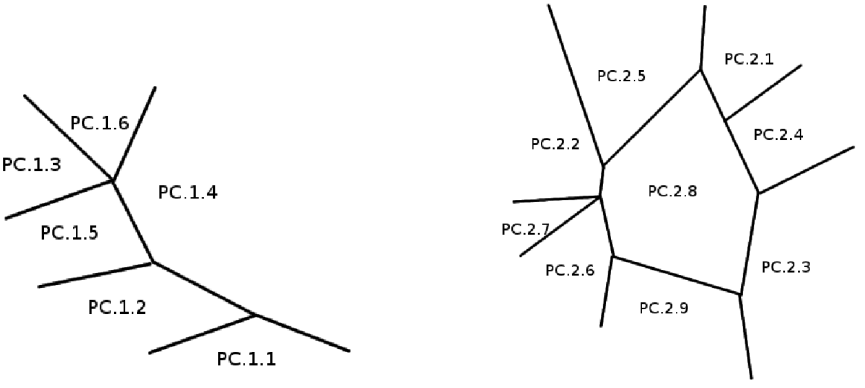
In this paper, we explore the possibility of using feature clustering for selecting spatial areas of interest in the WIPER system. We examine the distance between the time series of neighboring postal codes using the correlation distance described above and Euclidean distance, and we use these distance measures in conjunction with the single-link algorithm to visualize changes in the relationship between the call activities in neighboring postal codes when an emergency event occurs.

3 Experimental Setup

We examine the expression of two emergency events in real-world cell phone usage data. The first emergency event is an explosion, and the second is a riot. The two emergencies occur in different geographic locations and take place at different times of the year. First, we establish that each emergency event produces a corresponding change in the service usage and that the impact of the event on the call activity decreases as an increasing area surrounding the emergency event is considered. We aggregate the call data to count the number of phone calls made in a set of postal codes every 10 minutes in the city in which the emergencies occur. We study the two postal codes in which the emergency events occur and their neighbors. Next, we examine the correlation distance and Euclidean distance between the call activity time series of the postal codes in which the emergencies occur and the neighboring postal codes. Finally, we cluster the call activities time series for each set of postal codes for a normal day and the day of the emergency using an agglomerative clustering algorithm.

To measure the effect of the emergency events on the call activity of area surrounding the event, we first determine the location of the event from news reports. Using geographic information system tools, we establish an approximate latitude and longitude of the event. With this information and the latitude and longitude of the towers, we can filter the data to obtain calls made within any desired radius around the event.

For the remaining work, we aggregate the call activity by postal code. For each emergency event, we examine the postal codes containing the approximate latitude and longitude established for each emergency and their neighboring postal codes in the city in which the emergency occurs. There are 6 postal codes surrounding the first emergency (an explosion) and 9 surrounding the second (a riot). We denote the postal codes for each emergency as PC.1.1, PC.1.2, ..., PC.1.6 and PC.2.1, PC.2.2, ... PC.2.9, respectively. The first emergency occurs in PC.1.4, and the second occurs in PC.2.8. Figure 1 shows the approximate configuration of the postal codes.



(a) Emergency 1 **(b)** Emergency 2

Fig. 1. Approximate configuration of the postal codes. Emergency event 1 occurs in PC.1.4, which is at the edge of the city. Emergency event 2 occurs in PC.2.8, which is in the center of the city.

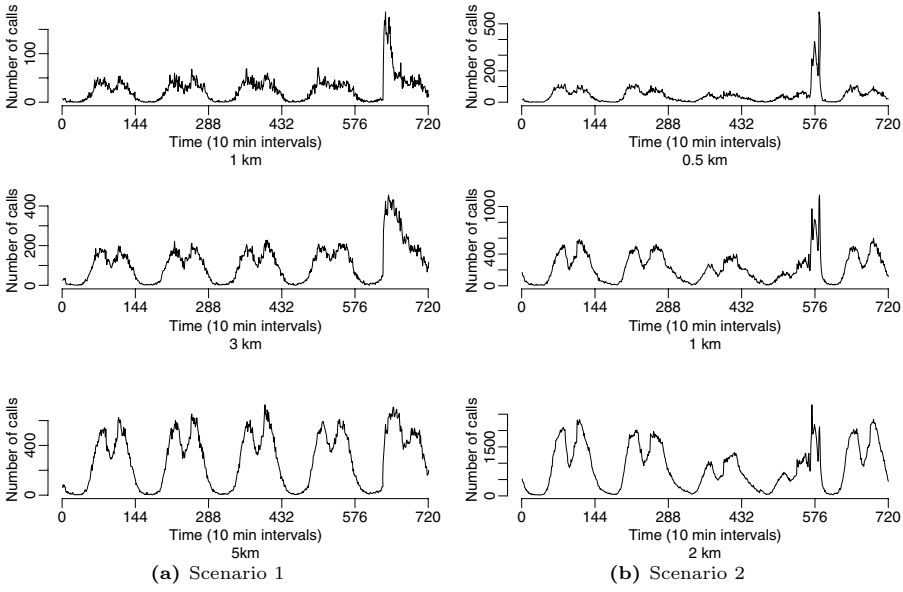
For each emergency event, we examine the correlation distance and the Euclidean distance between the postal code in which the emergency occurs and the neighboring postal codes for two weeks leading up to each event. We examine both the cumulative correlation distance and utilize a sliding window. Euclidean distance is computed only over a sliding window of 1 day of data. Both sliding windows contain the most recent 144 observations (taken at 10 minute intervals).

Finally, we compare time series clusterings for the two emergency events with those of normal call activity. We use single link agglomerative clustering with correlation distance and Euclidean distance, and we visualize the clusters using dendrograms.

4 Results

The columns in Fig 2 show the time series of call activities for the five days leading up to each emergency. Each row, from the top of the figure to the bottom, includes data from a greater area surrounding the location of the emergencies. The first emergency (left column) occurs at about 11 A.M. on the fifth day, and we see a corresponding increase in call activity at this time (approximately 640 minutes). The severity of this spike in activity decreases as the radius of the area increases from 1 km to 5 km. The second emergency (right column) occurs at approximately 2 o'clock on the morning of the fifth day, though we see elevated call activity even before midnight. As with the first scenario, the spike in call activity becomes less dramatic as a larger area, up to 2 km in radius, surrounding the emergency is included.

Figures 3 and 4 each show the correlation (both cumulative and over a sliding window) and Euclidean distances between the postal codes in which the emergency events occur and the neighboring postal codes for two weeks leading



(a) Scenario 1 **(b)** Scenario 2

Fig. 2. This figure shows the effect of the emergency situations on the call activity through the surrounding cell towers. The left column shows the time series for the first situation, which occurs at approximately 11 A.M. on the fifth day in the time series (approximately 642 minutes). The right column shows the time series for the second situation, which occurs at approximately two o'clock on the fifth day (approximately 588 minutes). In both cases, the severity of the activity spike decreases as a greater area is considered.

up to the emergency events. The left columns show the cumulative correlation distance used by Rodrigues *et al.* [13]. The center and right columns show the correlation distance and Euclidean distance, respectively, over a one day sliding window.

In Fig 3, we see an increase in each distance measure at the end of each time series. The cumulative correlation distance has only a slight increase at the end of the time series when the emergency event happens. These increases are more dramatic in the cases where a sliding window is used. Note that in the time series of Fig 3 there are two days of missing data, from 576 to 864 minutes. These missing data are not noticeable in the cumulative correlation distance; however, they lead to undefined correlation distances and Euclidean distances of 0 for 144 time steps when the entire sliding window contains 0 for all features. In Fig 4 we see similar increases in distance. The fact that the cumulative correlation distance shows only a small increase compared to the case where only a portion of the history of the time series is considered may indicate that this distance measure is dominated by older observations, making this cumulative measure to insensitive anomalies. The detrimental affect of old, stale data is discussed by Aggarwal in [14].

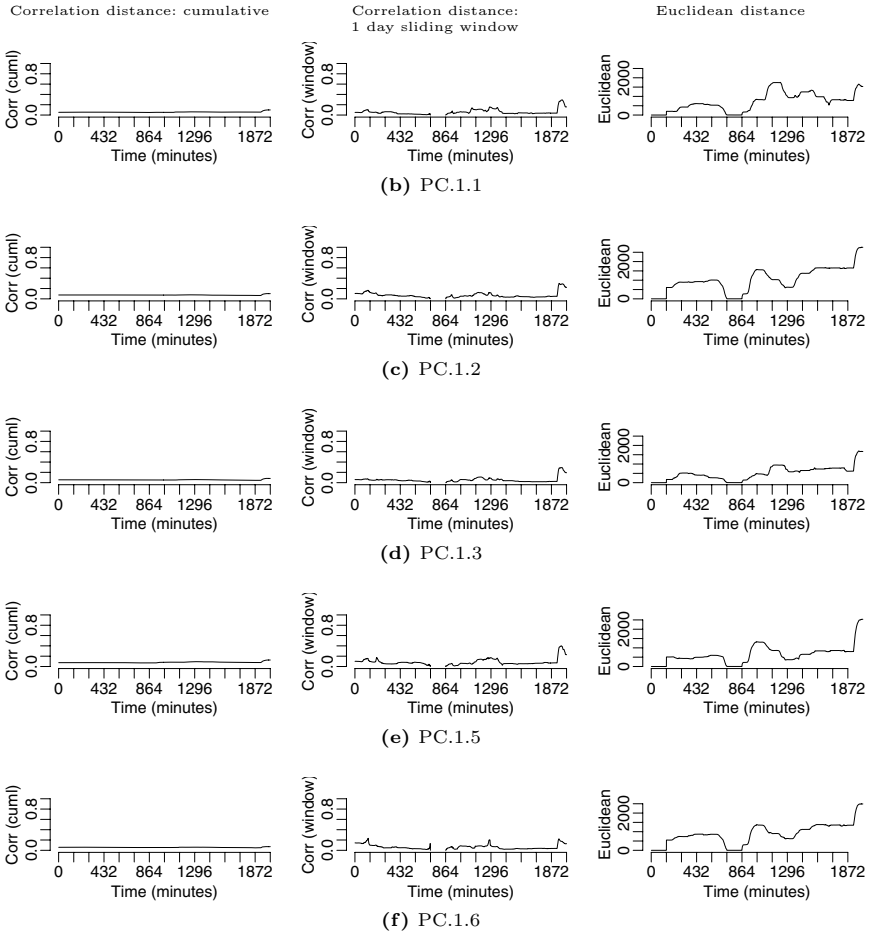
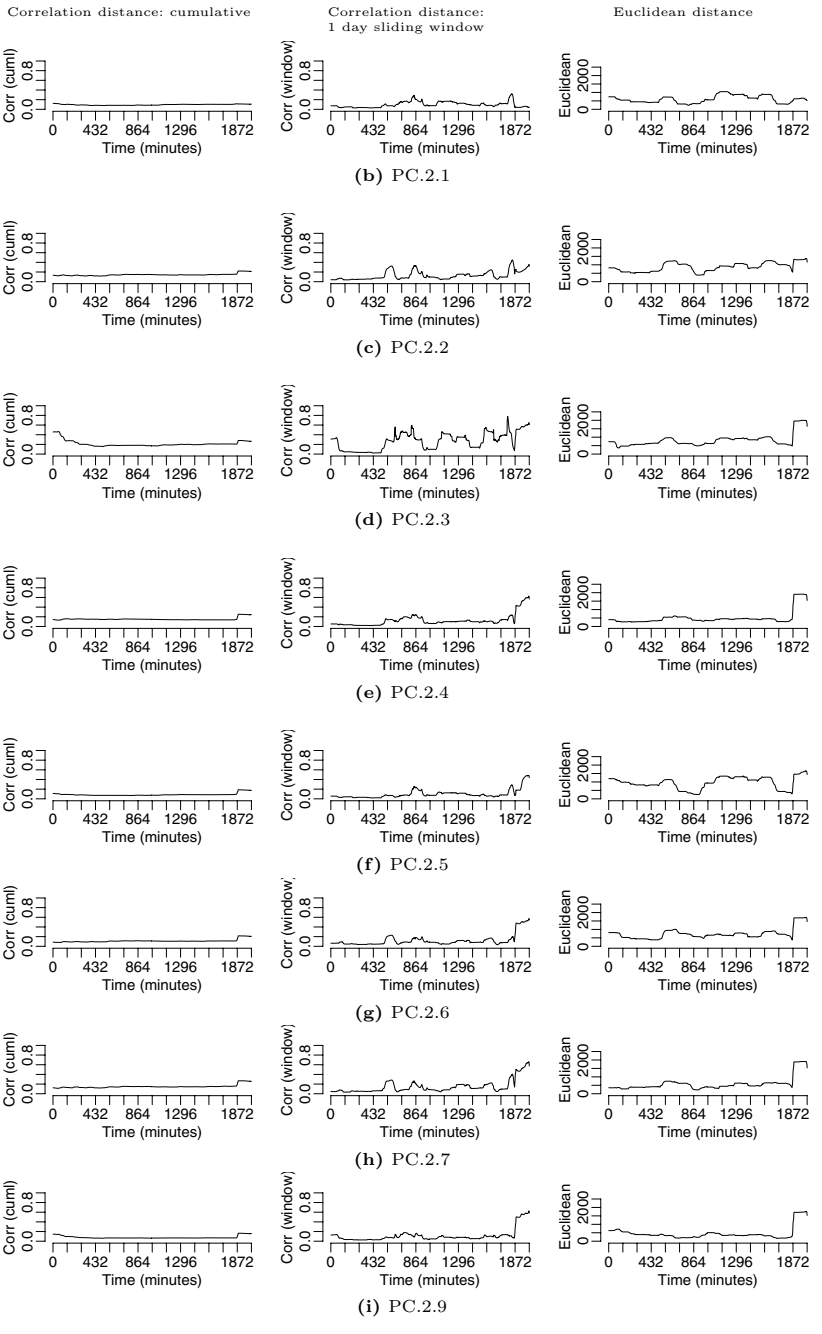


Fig. 3. This figure shows the correlation distance (cumulative and sliding window) and Euclidean distance between the postal code in which emergency event 1 occurs (PC.1.4) and its five neighboring postal codes for a two week period leading up to the emergency situation

Figures 5 and 6 show the contrast in clusterings for a day of normal activity (left column) and a day containing an emergency (right column). We cluster each day of data with the single link agglomerative algorithm using two different dissimilarity measures: correlation distance and Euclidean distance. Figure 5 shows the clusters for the first emergency situation. In both the correlation and Euclidean distance clusterings, the distance of PC.1.4, the postal code in which the emergency occurred, is significantly larger than the distance between any two clusters on the day of normal activity. In Fig 6, we see a similar separation of PC.2.8, the postal code in which the emergency occurred, along with PC.2.1



(b) PC.2.1

(c) PC.2.2

(d) PC.2.3

(e) PC.2.4

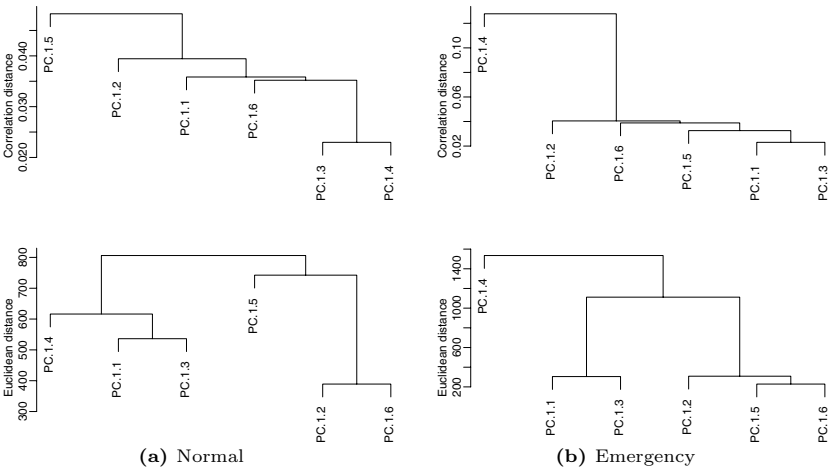
(f) PC.2.5

(g) PC.2.6

(h) PC.2.7

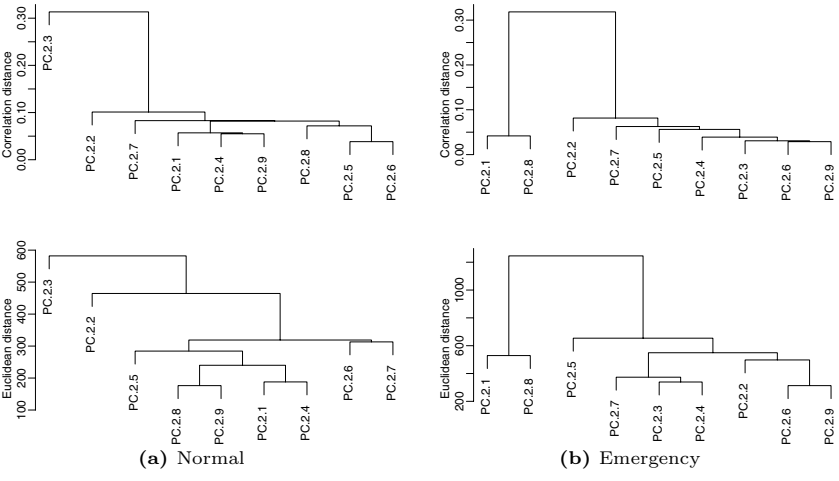
(i) PC.2.9

Fig. 4. This figure shows the correlation distance (cumulative and sliding window) and Euclidean distance between the postal code in which emergency 2 occurs (PC.2.8) and its five neighboring postal codes for a two week period leading up to the emergency situation



(a) Normal **(b)** Emergency

Fig. 5. This figure shows the clustering of the call volume time series for postal codes surrounding the first emergency event. The left column shows the clustering of one day of normal activity and the right column shows the clustering of the day of the first emergency event, which occurs in PC.1.4. Note that for both the correlation distance (top row) and Euclidean distance (bottom row), PC.1.4 is near other clusters during the day of normal activity but far from all other clusters during the day of the event.



(a) Normal **(b)** Emergency

Fig. 6. This figure shows the clustering of the call volume time series for postal codes surrounding the second emergency event. The left column shows the clustering of one day of normal activity and the right column shows the clustering of the day of the second emergency event, which occurs in PC.2.8. In this case, the call activity in PC.2.1 is also affected by this event. Note that for both the correlation distance (top row) and Euclidean distance (bottom row), the cluster containing PC.2.1 and PC.2.8 is near other clusters during the day of normal activity but far from all other clusters during the day of the event.

from the remaining clusters, though the increase in distance is not as dramatic as in the previous case. It is not surprising that PC.2.1 and PC.2.8 end up in the same cluster on the day of the emergency since we do not see the same increase in distance in Fig 4 between these to features as we do between PC.2.8 and the remaining features at the time the emergency occurs.

5 Conclusions and Future Work

In this paper, we have explored the possibility of using feature clustering to identify areas of interest from a set of spatially disjoint time series from real-world cell phone data. We have shown that emergency events can cause a spatially constrained change in call activity and that the area affected by this change can be detected using a clustering algorithm.

While this approach is promising, there is more work to be done before it can be deployed in the WIPER system. We need to determine the appropriate parameters for the approach, including the time series sampling interval, the level of spatial aggregation, and the length of the sliding window. Most importantly, while the dendrograms we have presented are compelling, we must do more work to understand how the clusters change over time in the absence of emergency events to gain an understanding of their stability and the amount of variation to be expected under normal circumstances. The work in this paper has been mostly qualitative, we must now pursue a more quantitative approach to automate the detection of areas of interest using feature clustering.

References

1. Madey, G.R., Barabási, A.L., Chawla, N.V., Gonzalez, M., Hachen, D., Lantz, B., Pawling, A., Schoenharl, T., Szabó, G., Wang, P., Yan, P.: Enhanced situational awareness: Application of DDDAS concepts to emergency and disaster management. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 1090–1097. Springer, Heidelberg (2007)
2. Pawling, A., Schoenharl, T., Yan, P., Madey, G.: WIPER: An emergency response system. In: Fiedrich, F., de Walle, B.V. (eds.) Proceedings of the 5th International ISCRAM Conference (2008)
3. Pawling, A., Yan, P., Candia, J., Schoenharl, T., Madey, G.: Anomaly Detection in Streaming Sensor Data. In: Intelligent Techniques for Warehousing and Mining Sensor Network Data. IGI Global (forthcoming)
4. Madey, G.: WIPER: The Integrated Wireless Phone-based Emergency Response System (2008), `http://www.nd.edu/~dddas`
5. Schoenharl, T.W., Madey, G.: Evaluation of measurement techniques for the validation of agent-based simulations against streaming data. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 6–15. Springer, Heidelberg (2008)
6. Darema, F.: Dynamic data driven applications systems: A new paradigm for application simulations and measurements. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3038, pp. 662–669. Springer, Heidelberg (2004)

7. Douglas, C.C.: DDDAS: Dynamic Data Driven Application Systems (2008), http://www.dddas.org
8. Plale, B., Gannon, D., Reed, D., Graves, S., Droegemeier, K., Wihelmson, B., Ramamurthy, M.: Towards dynamically adaptive weather analysis and forecasting in LEAD. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3515, pp. 624–631. Springer, Heidelberg (2005)
9. Flikkema, P.G., Agarwal, P.K., Clark, J.S., Ellis, C., Gelfand, A., Munagala, K., Yang, J.: From data reverence to data relevance: Model-mediated wireless sensing of the physical environment. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 988–994. Springer, Heidelberg (2007)
10. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1189 (2003)
11. Dhillon, I.S., Mallela, S., Kumar, R.: A divisive information-theoretic feature clustering algorithm for text classification. Journal of Machine Learning Research 3, 1265–1287 (2003)
12. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: A review. ACM Computing Surveys 31(3), 264–323 (1999)
13. Rodrigues, P., Gama, J., Pedroso, J.P.: Hierarchical time-series clustering for datastreams. In: Proceedings of the First International Workshop on Knowledge Discovery in Data Streams (2004)
14. Aggarwal, C.C.: On biased reservoir sampling in the presence of stream evolution. In: Proceedings of the 32nd Conference on Very Large Databases (2006)

An Ensemble Kalman-Particle Predictor-Corrector Filter for Non-Gaussian Data Assimilation

Jan Mandel[1,2] and Jonathan D. Beezley[1,2]

[1] University of Colorado Denver, Denver, CO 80217-3364, USA
[2] National Center for Atmospheric Research, Boulder, CO 80307-3000, USA
{jon.beezley.math,jan.mandel}@gmail.com

Abstract. An Ensemble Kalman Filter (EnKF, the predictor) is used make a large change in the state, followed by a Particle Filer (PF, the corrector), which assigns importance weights to describe a non-Gaussian distribution. The importance weights are obtained by nonparametric density estimation. It is demonstrated on several numerical examples that the new predictor-corrector filter combines the advantages of the EnKF and the PF and that it is suitable for high dimensional states which are discretizations of solutions of partial differential equations.

Keywords: Dynamic data driven application systems, data assimilation, ensemble Kalman filter, particle filter, tracking, non-parametric density estimation, Bayesian statistics.

1 Introduction

Dynamic Data Driven Application Systems (DDDAS) [1] aim to integrate data acquisition, modeling, and measurement steering into one dynamic system. Data assimilation is a statistical technique to modify model state in response to data and an important component of the DDDAS approach. Models are generally discretizations of partial differential equations and they may have easily millions of degrees of freedom. The model equations themselves are posed in functional spaces, which are infinitely dimensional. Because of nonlinearities, the probability distribution of the state is usually non-Gaussian.

A number of methods for data assimilation exist [2]. Filters attempt to find the best estimate from the model state and the data up to the present. We present a combination of the Ensemble Kalman Filter (EnKF) [3] and the Sequential Importal Sampling (SIS) particle filter (PF) [4]. The EnKF is a Monte-Carlo implementation of the Kalman Filter (KF). The KF is an exact method for Gaussian distributions. However, it needs to maintain the state covariance matrix, which is not possible for large state dimension. The EnKF and its variants [6,7] replace the covariance by the sample covariance computed from an ensemble of simulations. Each ensemble member is advanced in time by the model independently until analysis time, when the data is injected, resulting in changes in the states of the ensemble members. Particle filters also evolve a

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 470–478, 2009.

ensemble of simulations, but they assign to each ensemble member a weight and the analysis step updates the weights.

The KF and the EnKF represent the probability distributions by the mean and the covariance, and so they assume that the distributions are Gaussian. This shows in the tendency of EnKF to smear distributions towards unimodal, as illustrated in Sec. 3 below. So, while the EnKF has the advantage that it can make large charges in the state and the ensemble can represent an arbitrary distribution, the EnKF is still essentialy limited to Gaussian distributions. On the other hand, the PF can represent non-Gaussian distributions faithfully, but it only updates the weights and cannot move ensemble members in the state space. Thus a method that combines the advantages of both without the disadvantages of either is of interest. The design of more efficient non-Gaussian filters for large-scale problems has been the subject of significant interest, often using Gaussian mixtures and related approaches [8].

The predictor-corrector filter presented here uses an EnKF as a predictor to move the state distribution towards the correct region and then a PF as corrector to adjust for a non-Gaussian character of the distribution. Nonparametric density estimation is used to compute the weights in the PF. The combined predictor-corrector method appears to work well on problems where either EnKF of PF fails, and it does not degrade the performace of the EnKF for Gaussian distributions. Predictor-corrector filters were first formulated in [9,10]. Related results and some probabilistic background can be found in [11].

2 Formulation of the Method

A common procedure to construct an initial ensemble is as a sum with random coefficients [12],

$$u = \sum_{n=1}^{m} \lambda_n d_n \varphi_n, \quad d_n \sim N(0,1), \quad \{d_n\} \text{ independent}, \tag{1}$$

where $\{\varphi_n\}$ is an orthonormal basis in the space state $V = \mathbb{R}^m$ equipped with the Euclidean norm $\|\cdot\|$. The elements of V are column vectors of values of functions on a mesh in the spatial domain. The basis functions φ_n are smooth for small n and more oscillatory for large n. If the the coefficients $\lambda_n \to 0$ sufficiently fast, the series (1) converges and u is a random smooth function in the limit as $m \to \infty$. The sum (1) defines a Gaussian random variable with the eigenvalues of its covariance matrix equal to λ_k^2. Possible choices of $\{\varphi_k\}$ include a Fourier basis, such as the sine or cosine functions, or bred vectors [2]. On the state space V, we define another norm by

$$\|u\|_U^2 = \sum_{n=1}^{m} \frac{1}{\kappa_n^2} c_n^2, \quad u = \sum_{n=1}^{m} c_n \varphi_n. \tag{2}$$

Note that if $\kappa_n = 1$, $\|\cdot\|_U$ is just the original norm $\|\cdot\|$ on V. We generally use κ_n adapted to the smoothness of the functions in the initial ensemble, $\lambda_n/\kappa_n \to 0$ as $n \to \infty$.

A weighted ensemble of N simulations $(u_k, w_k)_{k=1}^N$ is initialized according to (1), with equal weights $w_k = 1/N$. The ensemble members are advanced by the model and at given points in time, new data is injected by an *analysis step*. The data consists of vector d of measurements, observation function $h(u) = Hu$, also called forward operator, here assumed to be linear, which links the model state space with the data space, and data error distribution, here assumed to be Gaussian with zero mean and known covariance R. The value of the observation function Hu is what the data vector would be in the absence of model and data errors. The value of the probability density of the data error distribution at the data vector d for a given value of the observation function Hu is called *data likelihood* and denoted by $p(d|u)$. The probability distribution of the model state before the data is injected is called the *prior* or the *forecast*, and the distribution after the data is injected is called the *posterior* or the *analysis*. Assuming the forecast probability distribution has the density p^f, the density p^a of the analysis is found from the Bayes theorem,

$$p^a(u) \propto p(d|u)\, p^f(u), \tag{3}$$

where $\propto$ means proportional.

Instead of working with densities, the probability distributions are approximated by weighted ensembles. We will call the following analysis step algorithm *EnKF-SIS*.

Predictor. Given a forecast ensemble

$$\left(u_k^f, w_k^f\right)_{k=1}^N, \quad w_k^f \geq 0, \quad \sum_{k=1}^N w_k^f = 1,$$

the members u_k^a of the analysis ensemble are found from the EnKF,

$$u_k^a = u_k^f + K(d_k - Hu_k^f), \quad d_k \sim N(d, R), \quad K = QH^{\mathrm{T}}\left(HQH^{\mathrm{T}} + R\right)^{-1}$$

where d_k are randomly sampled from the data distribution, and Q is the forecast ensemble covariance,

$$Q = \sum_{k=1}^N w_k \left(u_k - \overline{u}^f\right)\left(u_k - \overline{u}^f\right)^{\mathrm{T}}, \quad \overline{u}^f = \sum_{k=1}^N w_k^f u_k^f. \tag{4}$$

This is the EnKF from [3], extended to weighted ensembles by the use of the weighted sample covariance (4).

Corrector. The analysis members u_k^a are thought of as a sample from some *proposal distribution*, with density p^p. Ideally, the analysis weights w_k^a should be computed from the SIS update as [4]

$$w_k^a \propto p(d|u_k^a)\, \frac{p^f(u_k^a)}{p^p(u_k^a)}.$$

However, the ratio of the densities is not known, so it is replaced by a noparametric estimate inspired by [13], giving

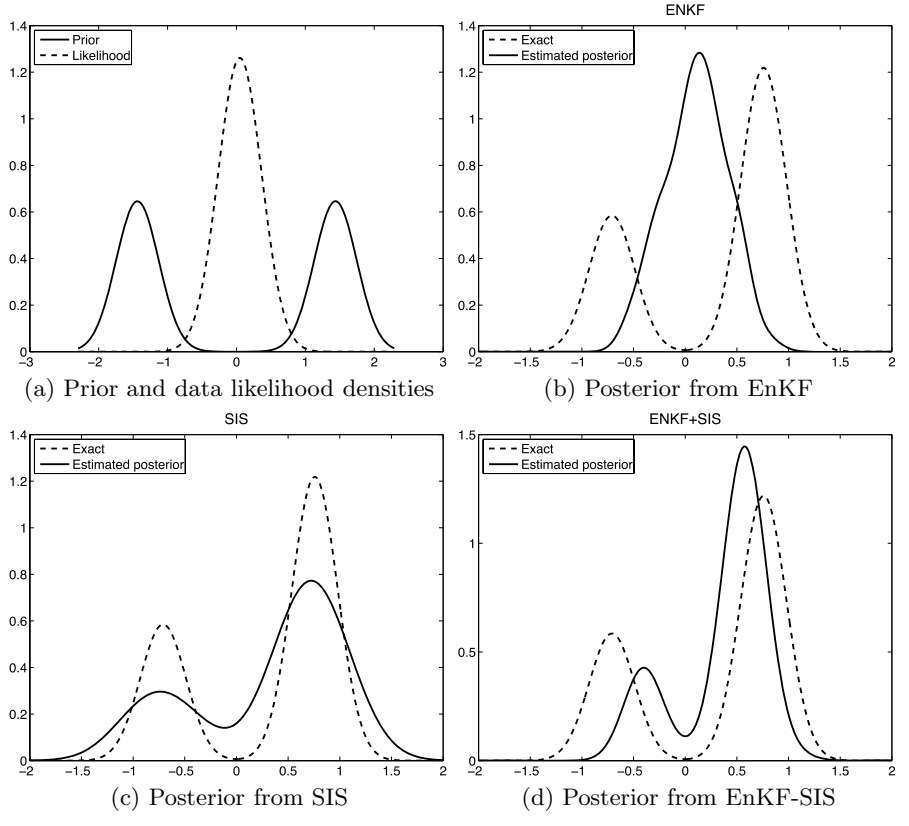
$$w_k^a \propto p\left(d|u_k^a\right) \frac{\sum_{\ell:\left\|u_\ell^f - u_k^a\right\|_U \leq h_k} w_k^f}{\sum_{\ell:\left\|u_\ell^a - u_k^a\right\|_U \leq h_k} \frac{1}{N}}, \quad \sum_{k=1}^{N} w_k^a = 1.$$

The bandwidth h_k is the distance from u_k^a to the $\lfloor N^{1/2} \rfloor$-th nearest member u_ℓ^a, measured in the $\|\cdot\|_U$ norm.

3 Numerical Results

Figure 1 demonstrates a failure of EnKF for non-Gaussian distributions, while SIS and EnKF-SIS do fine. We construct a bimodal prior in 1D by first sampling from $N(0, 5)$ and assigning the weights by

$$w_f(x_i) = e^{-5(1.5-x_i)^2} + e^{-5(-1.5-x_i)^2}.$$



(a) Prior and data likelihood densities (b) Posterior from EnKF

(c) Posterior from SIS (d) Posterior from EnKF-SIS

Fig. 1. Data assimilation with bimodal prior. EnKF fails to capture the non-Gaussian features of the posterior, but both SIS and EnKF-SIS represent the nature of the posterior reasonably well.

The data likelihood is Gaussian. The ensemble size was $N = 100$.

The next 1D problem demonstrates that EnKF-SIS is doing better that either EnKF or SIS alone in filtering for the stochastic differential equation [14]

$$\frac{du}{dt} = 4u - 4u^3 + \kappa\eta, \tag{5}$$

where $\eta(t)$ is white noise. The parameter κ controls the magnitude of the stochastic term.

The deterministic part of this differential equation is of the form

$$\frac{du}{dt} = -f'(u),$$

where the potential $f(u) = -2u^2 + u^4$. The equilibria are given by $f'(u) = 0$; there are two stable equilibria at $u = \pm 1$ and an unstable equilibrium at $u = 0$. The stochastic term of the differential equation makes it possible for the state to flip from one stable equilibrium point to another; however, a sufficiently small κ insures that such an event is rare.

A suitable test for an ensemble filter will be to determine if it can properly track the model as it transitions from one stable fixed point to the other. From Fig. 1, it is clear that EnKF will not be capable of describing the bimodal nature

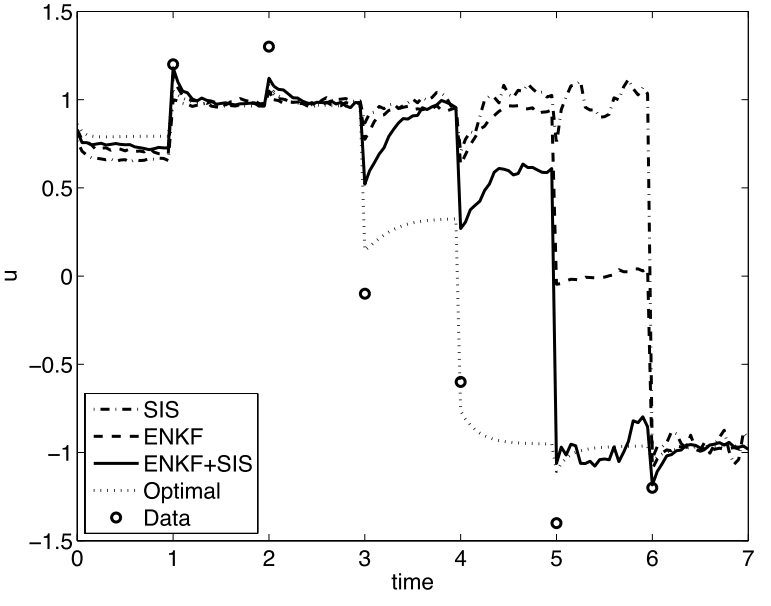


Fig. 2. Ensemble filters mean and optimal filter mean for stochastic ODE (5). EnKF-SIS was able to approximate the optimal solution better than either SIS or EnKF alone.

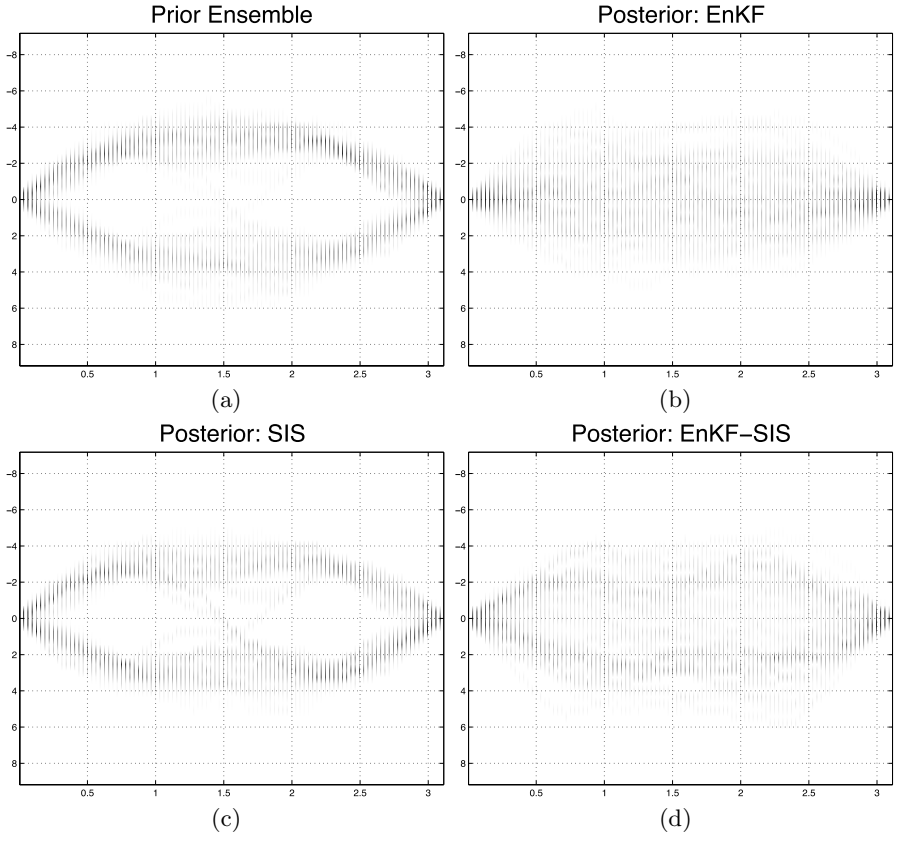


Fig. 3. EnKF smears non-Gaussian distribution. The horizontal axis is the spatial coordinate $x \in [0, \pi]$. The vertical axis is the value of function u. The level of shading on each vertical line is the marginal density of u at a fixed x, computed from a histogram with 50 bins. While EnKF completely ignores the non-Gaussian character of the posterior and centers the distribution around $u = 0$, EnKF-SIS shows darker bands at the edges.

of the state distribution so it will not perform well when tracking the transition. Also, when the ensemble is centered around one stable point, it is unlikely that some ensemble members would be close to the other stable point. It is known that SIS can be very slow in tracking the transition and EnKF can do better [14]. Figure 2 demonstrates that EnKF can outperform both.

The solution u of (5) is a random variable dependent on time, with density $p(t, u)$. The evolution of the density in time is given by the Fokker-Planck equation, which was solved numerically on a uniform mesh from $u = -3$ to $u = 3$ with the step $\Delta u = 0.01$. At the analysis time, the optimal posterior density was computed by multiplying the probability density of u by the data likelihood following (3) and then scaling so that again $\int p du = 1$, using numerical

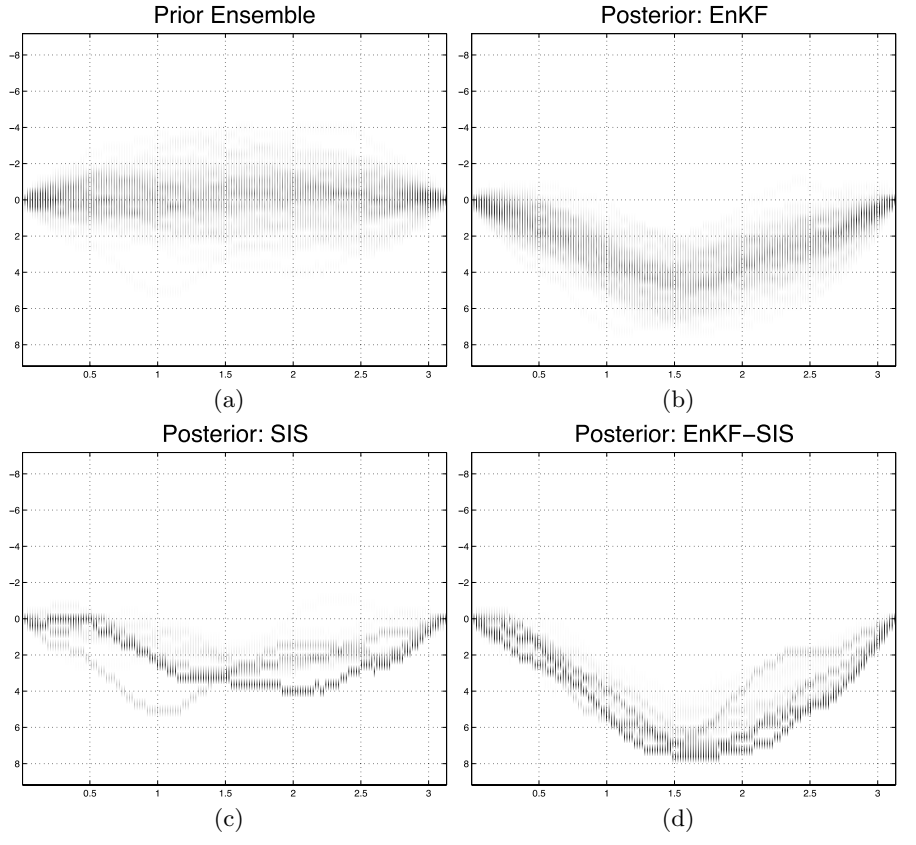


Fig. 4. SIS cannot make a large update. The horizontal axis is the spatial coordinate $x \in [0, \pi]$. The vertical axis is the value of function u. The level of shading on each vertical line is the marginal density of u at a fixed x, computed from a histogram with 50 bins. While EnKF and EnKF-SIS create ensembles that are attracted to the data value $u(\pi/2) = 7$, SIS cannot reach so far because there are no such members in this relatively small ensemble of size $N = 100$.

quadrature by the trapezoidal rule. The data points were taken from one solution of this model, called a reference solution, which exhibits a switch at time $t \approx 1.3$. The data error distribution was normal with the variance taken to be 0.1 at each point. To advance the ensemble members and the reference solution, we have solved (5) by the explicit Euler method with a random perturbation from $N(0, (\Delta t)^{1/2})$ added to the right hand side in every step [16]. The simulation was run for each method with ensemble size 100, and assimilation performed for each data point.

Finally, typical results for filtering in the space of functions on $[0, \pi]$ of the form

$$u = \sum_{n=1}^{d} c_n \sin(nx) \tag{6}$$

are in Figs. 3 and 4. The ensemble size was $N = 100$ and the dimension of the state space was $d = 500$. The Fourier coefficients were chosen $\lambda_n = n^{-3}$ to generate the initial ensemble from (1), and $\kappa_n = n^{-2}$ for the norm in the density estimation.

Figure 3 again shows that EnKF cannot handle bimodal distribution. The prior was constructed by assimilating the data likelihood

$$p(d|u) = \begin{cases} 1/2 \text{ if } u(\pi/4) \text{ and} \\ \quad u\,(3\pi/4) \in (-2,-1) \cup (1,2) \\ 0 \text{ otherwise} \end{cases}$$

into a large initial ensemble (size 50000) and resampling to the obtain the forecast ensemble size $N = 100$ with a non-Gaussian density. Then the data likelihood $u\,(\pi/2) - 0.1 \sim N(0,1)$ was assimilated to obtain the analysis ensemble.

Figure 4 shows a failure of SIS. The prior ensemble sampled from Gaussian distribution with coefficients $\lambda_n = n^{-3}$ using (1) and (6), and the data likelihood was $u\,(\pi/2) - 7 \sim N(0,1)$.

4 Conclusion

We have demonstrated the potential of a predictor-corrector filter to perform a successful Bayesian update in the presence of non-Gaussian distributions, large number of degrees of freedom, and large change of the state distribution. Open questions include convergence of the filter in high dimension when applied to multiple updates over time, mathematical convergence proofs for the density estimation and for the Bayesian update, and performance of the filters when applied to systems with a large number of different physical variables and modes, as is common in atmospheric models.

Acknowledgements

This work was supported by the National Science Foundation under grants CNS-0719641, ATM-0835579, and CNS-0821794.

References

1. Darema, F.: Dynamic data driven applications systems: A new paradigm for application simulations and measurements. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3038, pp. 662–669. Springer, Heidelberg (2004)
2. Kalnay, E.: Atmospheric Modeling, Data Assimilation and Predictability. Cambridge University Press, Cambridge (2003)
3. Burgers, G., van Leeuwen, P.J., Evensen, G.: Analysis scheme in the ensemble Kalman filter. Monthly Weather Rev. 126, 1719–1724 (1998)
4. Doucet, A., de Freitas, N., Gordon, N. (eds.): Sequential Monte Carlo in Practice. Springer, Heidelberg (2001)

5. Evensen, G.: The ensemble Kalman filter: Theoretical formulation and practical implementation. Ocean Dynamics 53, 343–367 (2003)
6. Anderson, J.L.: An ensemble adjustment Kalman filter for data assimilation. Monthly Weather Rev. 129, 2884–2903 (1999)
7. Mitchell, H.L., Houtekamer, P.L.: An adaptive ensemble Kalman filter. Monthly Weather Rev. 128, 416–433 (2000)
8. Bengtsson, T., Snyder, C., Nychka, D.: Toward a nonlinear ensemble filter for high dimensional systems. J. of Geophysical Research - Atmospheres 108(D24), STS 2–1–10 (2003)
9. Mandel, J., Beezley, J.D.: Predictor-corrector ensemble filters for the assimilation of sparse data into high dimensional nonlinear systems. CCM Report 232, University of Colorado Denver (2006), http://math.ucdenver.edu/ccm/reports/rep232.pdf
10. Mandel, J., Beezley, J.D.: Predictor-corrector and morphing ensemble filters for the assimilation of sparse data into high dimensional nonlinear systems. In: 11th Symposium on Integrated Observing and Assimilation Systems for the Atmosphere, Oceans, and Land Surface (IOAS-AOLS), CD-ROM, Paper 4.12, 87th American Meteorological Society Annual Meeting, San Antonio, TX (2007), http://ams.confex.com/ams/87ANNUAL/techprogram/paper_119633.htm
11. Mandel, J., Beezley, J.D.: Predictor-corrector ensemble filters for data assimilation into high-dimensional nonlinear systems (2009) (in preparation)
12. Evensen, G.: Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. J. of Geophysical Research 99 (C5), 143–162 (1994)
13. Loftsgaarden, D.O., Quesenberry, C.P.: A nonparametric estimate of a multivariate density function. Ann. Math. Stat. 36, 1049–1051 (1965)
14. Kim, S., Eyink, G.L., Restrepo, J.M., Alexander, F.J., Johnson, G.: Ensemble filtering for nonlinear dynamics. Monthly Weather Rev. 131, 2586–2594 (2003)
15. Miller, R.N., Carter, E.F., Blue, S.T.: Data assimilation into nonlinear stochastic models. Tellus 51A, 167–194 (1999)
16. Higham, D.J.: An algorithmic introduction to numerical simulation of stochastic differential equations. SIAM Rev. 43, 525–546 (2001)

Computational Steering Strategy to Calibrate Input Variables in a Dynamic Data Driven Genetic Algorithm for Forest Fire Spread Prediction*

Mónica Denham, Ana Cortés, and Tomás Margalef

Departament d' Arquitectura de Computadors i Sistemes Operatius,
Universitat Autònoma de Barcelona, 08193 - Bellaterra (Barcelona) Spain

Abstract. This work describes a Dynamic Data Driven Genetic Algorithm (DDDGA) for improving wildfires evolution prediction. We propose an universal computational steering strategy to automatically adjust certain input data values of forest fire simulators, which works independently on the underlying propagation model. This method has been implemented in a parallel fashion and the experiments performed demonstrated its ability to overcome the input data uncertainty and to reduce the execution time of the whole prediction process.

1 Introduction

Forest fires are part of natural balance in our planet but, unfortunately, during last years the number of forest fires had increased in an alarming way. The high number of this kind of disasters break the natural balance that forest fire means.

Nowadays, people is arduously working on this problem in order to avoid and to reduce forest fires damages. As results of this effort there exist different kind of studies, strategies and tools used to prevent fires, to define risk areas and to reduce the fire effects when a disaster occurs.

Forest fire simulators are a very useful tool for predicting fire behavior, simulators allow us to know the fire progress, intensity, spread area, flame length, etc. Nowadays, there exist several forest fire simulators [7], which may differ in inputs, outputs, fire model implemented, fire type (crown, surfaces or underground fires), etc.

A forest fire simulator needs to be fed with data related to the environment where fire occurs such as terrain main features, weather conditions, fuel type, fuel load and fuel moistures, wind conditions, etc. However, it is very difficult to exactly evaluate the real time values of these parameters for different reasons. There are certain parameters that change through time such as air and fuel humidities. Environmental conditions are also affected by the fire itself due to its elevated temperatures, fires could generate very strong gust of winds as well,

* This work is supported by the MEC-Spain under contracts TIN 2007-64974.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 479–488, 2009.

etc. The lack of accuracy of the input parameter values adds uncertainty to the whole method and it usually provokes low quality simulations [1].

Thus, in order to achieve high simulation quality, our application is held at Dynamic Data Driven Application Systems (DDDAS) paradigm [4] [5] [9]. In particular, our prediction system explores multiple fire propagation scenarios (different combinations of the input parameters values) dynamically adapting those scenarios according to observed real fire evolution. By the observation of real fire progress, certain input parameter values are steered in order to reduce the whole search space achieving a response time reduction. Consequently, steering the parameter values will improve its value accuracy improving predictions quality as well.

Moreover, in order to reduce response time, we also had developed our application using a parallel solution (master/worker programming paradigm).

This work is organized as follow. Next section describes the proposed dynamic data driven forest fire prediction methodology compared to the classical prediction scheme. Section 3 is focused to describe the Calibration Stage of the proposed prediction methodology and two steering strategies are described. Experimental results are shown in section 4 and, finally, main conclusions and future work are reported in section 5.

2 Forest Fire Spread Prediction

Traditionally, forest fire prediction (figure 1 (a)) is carried out using a forest fire simulator (FS), a set of input parameters (slope, vegetation, dead fuel humidity, live fuel humidity, wind characteristics) and the state of the fire front at a given instant time t_i (called RFt_i: Real Fire for instant t_i). Using this information, the predicted fire line for a time t_{i+1} is obtained (SFt_{i+1}: Simulated Fire for instant t_{i+1}). This method consumes very few resources in terms of time and computation power (it performs just one simulation using the unique scenario).

However, this simplicity suffers from a very important drawback: usually the predictions obtained are far from the real fire spread due to simulator underlying uncertainty and the quality of the unique scenario.

In order to improve the prediction accuracy, we include a stage called Calibration Stage that will be executed before the classical prediction scheme (called Prediction Stage, figure 1 (b)). Within this new stage, we used a Genetic Algorithm (GA) for evolving a set of different scenarios in order to improve their values accuracy (each scenario is an individual of a given GA population).

At Calibration Stage, the beginning fire line (RFt_i) and a set of parameter values are used to obtain the simulated fire line for instant t_{i+1} (SFt_{i+1}). This simulated fire line and the real fire line (SF and RF both for instant t_{i+1}) are compared and the result of this comparison is used as feedback tunning information to improve parameter values accuracy. This process is executed for all population individual, through a prefixed number of generations. Once Calibration Stage ends, the Prediction Stage takes place where the best parameter's set values found in the Calibration Stage is used to feed the simulator and to obtain the final prediction for the next time instant (SFt_{i+2})) [1].

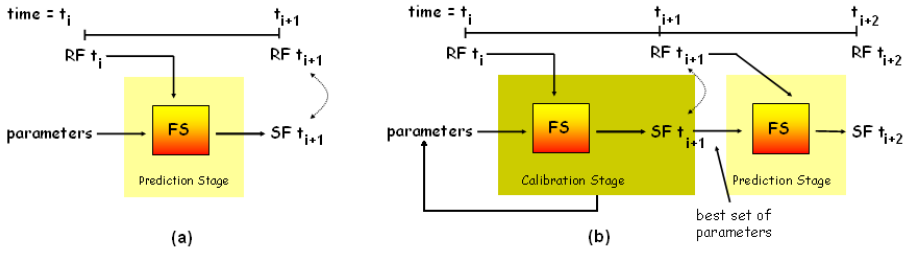


Fig. 1. (a) Classical method for forest fire prediction. (b) Two stage method for forest fire prediction.

Since in the Calibration Stage the original random values setting for the GA is updated to be driven by the observed real fire behavior, it could be referred as a Dynamic Data Driven Genetic Algorithm (DDDGA). In the subsequent section, the proposed DDDGA strategy is described.

3 Parallel Dynamic Data Driven Genetic Algorithm

Classical GAs [8] are inspired in evolutionary theory where a population of individuals is evolved generation by generation with the objective of improving population individuals characteristics. Some operations are applied to obtain each of the generations: *selection* (including elitism), *crossover* and *mutation*.

Through our application frame, the goodness of a given individual (scenario) is determined by real and simulated fire maps comparison (SFt_{i+1} and RFt_{i+1} in figure 1(b)) where maps differences determine the error of the simulation. Therefore, GA main goal is to minimize this error.

In this work, we use the forest fire simulator *fireLib* [3] [7] [10] for surface fires. Some of *fireLib* input parameters are: terrain slope (direction and inclination), wind speed and wind direction, moisture content of live fuel, moisture content of dead fuel (at three different times: 1 hour, 10 hours and 100 hours), vegetation type, etc. In particular, vegetation is modeled by considering the vegetation models defined in [2].

Vegetation type and slope are the most static parameters, therefore, their values will be considered static and known for each prediction process. Consequently, the DDDGA will only consider for evolving purposes the remainder parameters: wind direction and wind speed, moisture content of live fuel and the three moisture content of dead fuel.

In a previous work [6], the authors introduced the data assimilation process needed for this two stage prediction method and an analytical steering strategy for the Calibration Stage was also described.

It is well known that slope and wind are the two main features to determine fire progress. Thus, during most fire model implementations slope and wind factors are composed in order to obtain fire progress direction and velocity.

Through Calibration Stage, we dispose of fire progress from instant t_i to t_{i+1} (figure 1 (b)), then we analyze this fire progress and we obtain fire direction and velocity (the real fire spread characteristics for instant t_{i+1}).

Taking into account these facts and knowing slope characteristics (as we had mentioned in a previous paragraph), we could combine slope and real fire spread in order to obtain wind values, those which are necessary for achieving the observed fire spread [6]. For this purpose, an analytical steering strategy for the Calibration Stage was introduced in [6]. This approach shows good results reducing the error function and improving Calibration and Prediction Stages results. However, the main drawback exhibited by this steering method was its strong dependency to the underlying simulator. In order to overcome this penalty, we had developed a steering strategy called computational method, which has no dependence of the underlying fire simulator.

Next section will introduce the computational steering strategy main characteristics and, in subsequent sections, a comparative study of both analytical and computational steering strategies will be presented.

3.1 Computational Steering Strategy

The main advantage of the proposed computational steering strategy is its fire simulator independence. For this purpose, the underlying fire simulator is used as a black-box from which the only available information is the input/output data used/generated. Based on a complete set of fire spread information obtained from both real historical and synthetic fires, one generates a complete database of fire evolutions with the corresponding environmental conditions. This database information will be used for the DDDGA (Calibrationg stage) to discover the "ideal" wind values (wind speed and wind direction) based on a key search obtained from the real observed fire spread.

Figure 2 shows an example of how the DDDGA works under this computational steering strategy. Let's assume that the observed real fire at instant t_{i+1} exhibits a rate of spread equal to 20 fpm and a spread direction equal to 45^o. Furthermore, the slope is known and corresponds to 45^o (1 radian), and the observed fuel model is 7 [2]. The database register selected by the method is shown with a circle in the figure. This register has a wind speed equal to 9 fpm and its direction is 45^o azimuth. These wind values will be used to define a subrange through the whole parameter valid range and when *mutation* takes place, the wind values will be assigned using a random value limited by the new subrange (taking into account database cases incompleteness).

In order to validate the experimental results when the proposed computational steering strategy is applied, we have compared its results (computational method results) against analytical steering strategy results. Although we expected that analytical method performs better than the computational strategy, we wanted to demonstrate that the proposed method could reach a good performance prediction despite of not being aware of the underlying simulator model used.

Taking into account that fire simulation is the most time consuming task, we had proposed a master/worker solution for our parallel GA. Thus, master

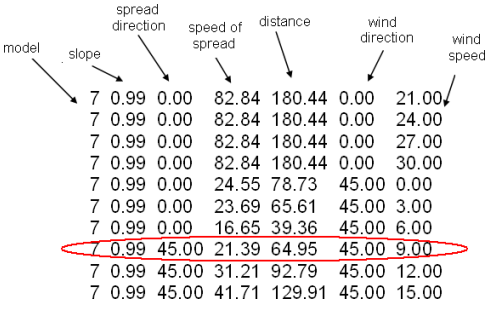


Fig. 2. Data stored in Computational Method data base

process holds the population, and it distributes the individuals among the worker processes and then, it receives the evaluated individuals (that means, each individual and its error). In turn, the worker processes receive a group of individuals (called chunk). For each individual of the received chunk, the worker process executes the fire spread simulation and evaluates the error function. When the worker finalizes with one chunk, it sends back to the master process the evaluated information. Then, the worker will keep waiting until another chunk of individuals is being received. This process is repeated until the whole population has been evaluated. Afterward, the master process applies genetic operators over the evaluated population in order to minimize the error value.

4 Experimental Results

As we had mentioned, two important key point in fire spread prediction are, on the one hand, to obtain the prediction results as fast as possible and, on the other hand, to provide simulation results as precise as possible. These two characteristics are essentials for having useful forest fire spread predictions. Thus, our experimentation covers these two topics: time reduction using parallel computing and error reduction when a dynamic data driven option is applied. Next sections describe how each experiment has been performed.

4.1 Parallel GA Performance Evaluation

The first experiment deals with application scalability. We had fixed a population size of 512 individuals (all generations have the same number of individuals), and we varied the number of workers from 1 to 31. A real map of 110 x 110 m^2 cells of $1m^2$ was used in this case, and we tested the Calibration Stage for a unique 8 minutes interval time. Figure 3 shows the time reduction when we execute the parallel Calibration Stage using computational steering method and no guided GA. In vertical axis we can see time (in seconds) and horizontal axis shows the number of workers used. The execution platform is a homogeneous PC cluster composed by 32 nodes. This cluster uses a queue system in order to guarantee exclusive access to the required resources.

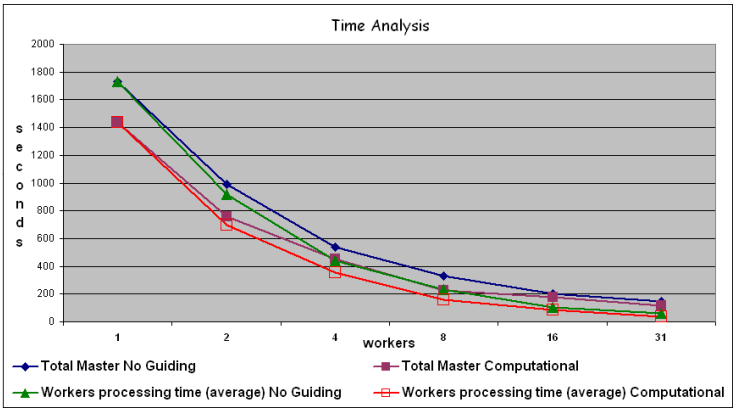


Fig. 3. Execution time varying the number of workers from 1 to 31

The application described in this work deals with a real problem, and the results provided by the proposed prediction scheme could be used for improving fire fighting actions. The final users of this application may be institutions such us civil protection and fire fighters, usually, can not directly access to big computational resources during real time hazards. From the results shown in figure 3, we can see that our application has an appropriate scalability when the number of workers is increased. In the next section, we report the experiments performed in order to evaluate the proposed steering methods.

4.2 Dynamic Data Driven GA Evaluation

In this section we are going to evaluate the benefit of applying a dynamic data driven GA for fire spread prediction purposes. We are going to compare the analytical method with the computational method applying the prediction methodology without considering any external data. In this work, we present some results using one real fire and two synthetic fires (simulation results). Figure 4 shows the fire front evolution through time (2 minutes intervals) for the three experiments.

In all cases, we had used a populations size of 50 individuals with random values at the beginning. Each population was evolved 5 times (error reduction is insignificant after the first 3 or 4 evolutions). The depicted results are the average of 5 different executions for each case using different initial populations. The initial populations were created by random values but the remainder time lapses previous evolved populations (at Calibration Stage) were used. Moreover, we use the best individual obtained after executing the Calibration Stage as input in the Prediction Stage.

Experiment 1: Synthetic Case. Experiment 1 concerns with the map shown in figure 4(a) (109.0 x 89 m^2, cells of $1m^2$). The terrain had 18^o slope and the

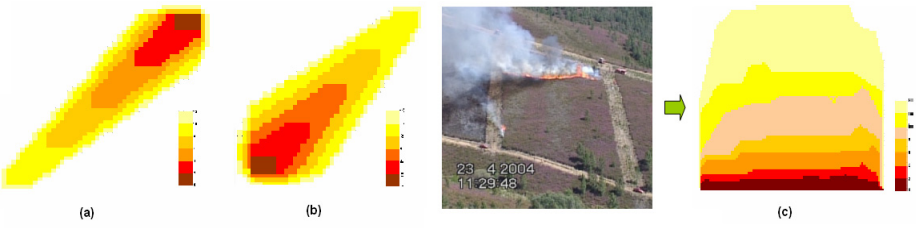


Fig. 4. (a) Experiment 1: synthetic fire case. (b) Experiment 2: synthetic fire case. (c) Experiment 3: real fire case.

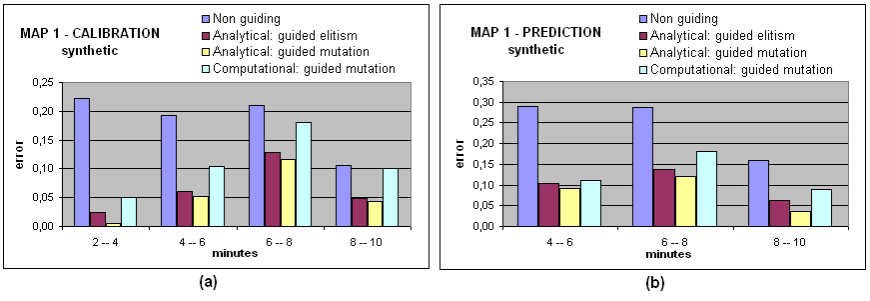


Fig. 5. Experiment 1: (a) Calibration Stage and (b) Prediction Stage results

vegetation fuel model was set up to model 7 [2]. Figures 5(a) and (b) show the Calibration and Prediction Stages results respectively.

For both stages (Calibration and Prediction), the error (difference between the real fire spread and the simulated fire obtained) has been significantly reduced whatever dynamic data driven methods was used. Since one of our goals was to use the analytical method as a validation element of the computational method, we have analyzed in more detail each method behavior. From an immediate analysis of graphics 5(a) and 5(b), we detect a clear similarity along all data driven methods. However, it is also remarkable that analytical method provides better results than the computational method as it was expected. Nevertheless, the error difference between the two analytical methods and the computational method keeps, on average, bounded by 25% for all interval times, therefore, we can conclude that the analytical results validate the computational behavior. Prediction Stage results show errors slightly higher than Calibration Stage, however this is an expected result because we are using the best individual obtained at the Calibration Stage to obtain the prediction for a later time interval at the Prediction Stage.

Experiment 2: Synthetic Case. Another synthetic map was used to carry out the second test (figure 4(b)), 33,22 x 27,12 m^2 size map, cells of 1 f^2. In this case, we consider 27^o slope and the same vegetation as in the first burning case.

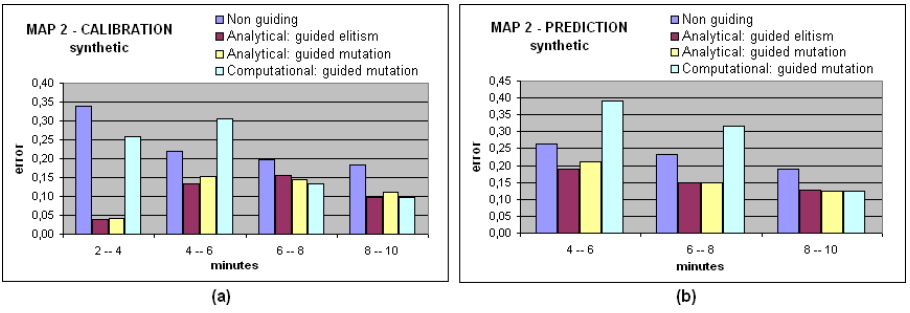


Fig. 6. Experiment 2: (a) Calibration Stage and (b) Prediction Stage results

Figure 6 shows the second experiment results. At Calibration Stage (figure 6 (a)) we can see that analytical method reduces the error in all cases. However, computational method has an unexpected behavior in steps 2-4 and 4-6 at Calibration Stage and 4-6, 6-8 at Prediction Stage. In order to understand this unexpected behavior we analyze each individual execution for those particular situations.

At Calibration Stage we observed that when computational method was used, the resulting error values of each population were similar. However, when no steering method were applied, some populations had generated high errors and another ones had generated very small errors. Thus, total average was under influence of high errors as well as small ones. Since the Prediction Stage results depend on Calibration Stage best individual quality, the behavior observed in the Calibration Stage was reflected in the Prediction Stage. Therefore, when we apply any of the proposed methods, we are avoiding to depend on "lucky" of choosing good random values. It is important to take into account that in disasters problems having stable algorithms means an important improvement if we can still guarantee good results.

In order to determine benefits of applying any steering methods and taking into account calibration stage (where steering methods are applied), we compare total error for each method (no guiding total error means 100%). When computational method was applied, error reduction was about 15,9% approximately, analytical method (guided elitism option) error reduction was about 55,3% and analytical method (guided mutation option) error reduction was about 52,1% approximately (all lapses time average). For the same reasons of experiment 1, we note that computational method behavior is validated by analitical method behaviors.

Experiment 3: Real Case. The last experiment is a real fire (figure 4(c)). The fire analyzed in this experiment corresponds to a plot of 89 x 109 m^2, 1 m^2 cell size. The terrain was 18^o slope and fuel type was equal 7. This burn has been extracted from a set of prescribe burns performed in the frame of an European project called SPREAD (Gestosa, 2004).

The results obtained for this experiment are shown in figure 7. The first 2 time intervals at Calibration Stage results were similar through all methods,

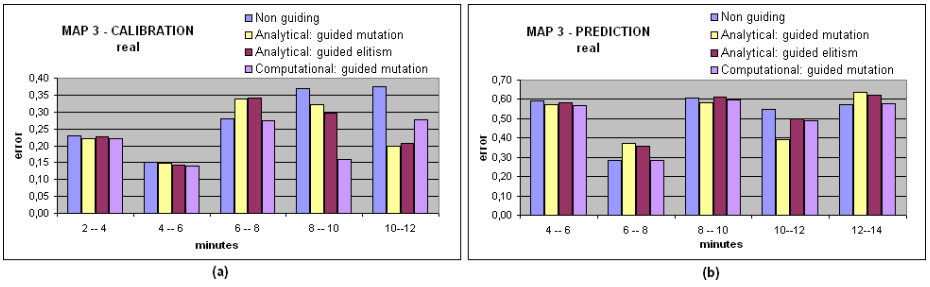


Fig. 7. Experiment 3: (a) Calibration Stage results and (b) Prediction Stage results

there were not significant differences between different configurations, however, this stable behavior changes from that point to the end. We can see that in experiment 3, the errors were larger compared to the previous experiment errors. This error improvement could be considered as an expected behavior because of being dealing with a real fire instead of a synthetic fire. The spread behavior, in this case, was not uniform because of vegetation, variable wind conditions, the fire itself, and so on. This non uniform environment is harder to be reproduced by any fire spread simulator. In this case, error reduction was, on average, for the computational method 23,5%, for the analytical method (guided elitism option) about 13,5% and, finally, for the analytical method when mutation was guided is on 12,1%.

5 Conclusions and Future Work

A Parallel Dynamic Data Driven Genetic Algorithm was proposed for forest fire spread prediction. This application deals with response time restrictions and prediction accuracy requirements. From the experimental results we could determine that our master/worker scheme was appropriate to take advantages of parallel computing in order to reduce the forest fire prediction response time.

On the other hand, in order to improve prediction accuracy, a Dynamic Data Driven GA was proposed where real fire progress was used for adapting the scenarios used by the method according to the observed real fire spread. A new steering method called Computational Steering method has been proposed, which main feature is being independent on the underlying fire simulator becoming an universal method for calibrating the input parameters of any fire spread simulator.

Three cases of study were presented and their results had shown that the inclusion of the dynamic data driven systems bases in the Calibration Stage improves the quality of the propagation predictions. Furthermore, since the analytical and the computational methods have a similar behavior that is bounded by a constant difference (around 25%), we can conclude that the computational method behavior does not exhibit unexpected characteristics. Thus, computational method development is validated by the analytical method results through our specific domain.

Although our main objectives are real burning maps, synthetic cases help us to validate the proposed methods. These cases prove that dynamic data driven GA improves the final results by reducing search space and avoiding simulations with individuals that because of their characteristics provide low quality simulations.

References

1. Abdalhaq, B.: A methodology to enhance the Prediction of Forest Fire Propagation. Ph.D Thesis. Universitat Autònoma de Barcelona (Spain) (June 2004)
2. Anderson, H.E.: Aids to Determining Fuel Models For Estimating Fire Behavior. Intermountain Forest and Range Experiment Station Ogden, UT 84401. General Technical Report INT.122 (1982)
3. Bevins C. D.: FireLib User Manual & Technical Reference (1996), `http://www.fire.org` (accessed, January 2006)
4. Darema, F.: Dynamic Data Driven Applications Systems: A New Paradigm for Application Simulations and Measurements. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3038, pp. 662–669. Springer, Heidelberg (2004)
5. Douglas, C.C.: Dynamic Data Driven Application Systems homepage, `http://www.dddas.org` (accessed, October 2008)
6. Denham, M., Cortés, A., Margalef, T., Luque, E.: Applying a Dynamic Data Driven Genetic Algorithm to Improve Forest Fire Spread Prediction. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 36–45. Springer, Heidelberg (2008)
7. FIRE.ORG - Public Domain Software for the Wildland fire Community, `http://www.fire.org` (accessed, May 2007)
8. Koza, J.: Genetic Programming. In: On the programming of computers by means of natural selection, Massachusetts Institute of Technology. Cambridge, Massachusetts 02142. The MIT Press, Cambridge (1992)
9. Mandel, J., Beezley, J., Bennethm, L., Chakraborty, S., Coen, J., Douglas, C., Hatcher, J., Kim, M., Vodacek, A.: A Dynamic Data Driven Wildladnd Fire Model. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 1042–1049. Springer, Heidelberg (2007)
10. Rothermel, R.C.: A mathematical model for predicting fire spread in wildland fuels. USDA FS, Ogden TU, Res. Pap. INT-115 (1972)

Injecting Dynamic Real-Time Data into a DDDAS for Forest Fire Behavior Prediction*

Roque Rodríguez, Ana Cortés, and Tomás Margalef

Departament d' Arquitectura de Computadors i Sistemes Operatius,
Universitat Autònoma de Barcelona, 08193 - Bellaterra (Barcelona) Spain
dario.rodriguez@caos.uab.es
{ana.cortes,tomas.margalef}@uab.es
http://www.caos.uab.es

Abstract. This work presents a novel idea for forest fire prediction, based on Dynamic Data Driven Application Systems. We developed a system capable of assimilating data at execution time, and conduct simulation according to those measurements. We used a conventional simulator, and created a methodology capable of removing parameter uncertainty. To test this methodology, several experiments were performed based on southern California fires.

Keywords: Dynamic Data Driven Application System, Parallel computing, Forest fire prediction, HPC, Evolutionary computing.

1 Introduction

Forest fires are one of the nature's most serious threats. Actually, there exist several tools for mitigating damages caused by fires such as fire propagation simulators, based in some physical or mathematical models, being Rothermel's the most recognized one [13]. However, most simulators of natural phenomena such as hurricanes and fires, are very computing demanding and they required as inputs a wide set of variables whose values are either not well known or estimated prior to execution including a considerable uncertainty degree. In fact, this static restrictions (variable inputs are set up only at the very beginning of the simulation process) is an important drawback because as the simulation time goes on, variables previously initialized could dramatically changed producing misleading simulations results. Therefore, to overcome these restrictions, we need a system capable of either obtaining or estimating the values of the input parameters needed by the underlying simulator correctly and, furthermore, this system must be able of adapting itself dynamically to the constant environment conditions changes, by means of real-time measurements. Those characteristics matches the definition of Dynamic Data Driven Application Systems (DDDAS)[5].

* This work is supported by the MEC-Spain under contract TIN 2007-64974.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 489–499, 2009.

Furthermore, nowadays there is a huge computer power available around the world because of distributed systems such as Grid environments and emerging technologies improvements such as multiple cores and new parallelization techniques. However, most of the current simulation tools are both off-line and sequential presenting slightly time restriction, such as most of the scientific applications. This work represents a step forward to make use of the available computing resources in order to drive this kind of applications to the dimension of the Urgent HPC applications [2].

In section 2, we describe the proposed prediction strategy $SAPIFE^3$ a two stage fire prediction method that overcomes time restrictions while reducing the skew in simulation results caused by sudden changes in the weather conditions. A brief description of the module responsible to inject data at run time is included in section 3. In section 4, we present the experimental study and, finally, the main conclusions are reported in section 5.

2 $SAPIFE^3$: A Two Stage Prediction Method

Our research team has proposed, in previous works, a paradigm change in forest fire prediction, coming from the classic prediction to DDDAS methods [3] [6]. The classic fire prediction scheme sets up only once the simulator's input variables at time instant t_0 (seen figure 1(a)) keeping them constant for the whole prediction phase (also called Prediction Stage). We include another phase, previous to the prediction one (called the Calibration stage), where the simulator's input parameters are calibrated, depending on the observed fire's behavior from t_0 to t_1. The calibrated values obtained at this Calibration stage will be used in the Prediction stage as it can be seen in figure 1(b).

In this work, we propose a two stage prediction scheme called $SAPIFE^3$, this is the spanish acronym for *Adaptive System for Fire Prediction Based in Statistical-Evolutive Strategies* (Sistema Adaptativo para la Prediccin de Incendios Forestales basados en Estratgias Estadstico-Evolutivas) [12]. This method couples two prediction schemes: a genetic algorithm and a statistical approach. Subsequently, both methods are described.

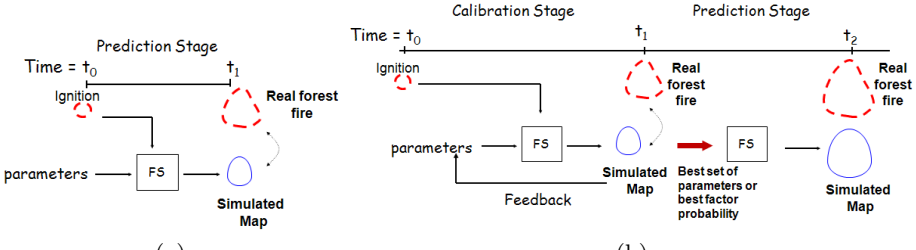


Fig. 1. Predicction Methods

Genetic Algorithm: This prediction method uses a Genetic Algorithm (GA) in the Calibration stage. The population used in the GA is composed of n individuals each of them being composed by a particular setting of the underlying simulator input parameters. We call each input parameter's combination a scenario. The $fitnessfunction$ is an error formula that returns the error between the real observed fire propagation map and the simulated propagation map. This error function will be evaluated for each scenario in order to rank them in terms of prediction quality. Since our system is based on a cell automaton, the error function used is the one defined in equation 1 where $InitCells$ are the cells where fire begun, $Cells \cup$ is the union between real and simulated fire spread, $Cells \cap$ is the intersection between real and simulated fire and $RealCells$ are the cells burnt by the real fire. Once all scenarios have been ranked, they will be updated according to elitism, selection, crossover and mutation operation and an improved population will be obtained. Once the evolution process (Calibration stage) is finished, the best population will be used in the statistical module.

$$Error = \frac{(Cells \cup -InitCells) - (Cells \cap -InitCells)}{RealCells - InitCells} \quad (1)$$

Statistical Integration: Originally, this method was called Statistical System for Forest Fire Management (S^2F^2M)[3]. This method is based on probabilities and has the aim of sweeping the whole search space exhaustively by considering almost all possible combinations of the simulator's input parameters. Obviously, this method generates a huge number of possible scenarios. When S^2F^2M is coupled to the GA, the number of scenarios used is reduced because it will receive as input population the one provided by the GA. Afterwards, S^2F^2M will evaluate the probability of any cell to be burnt or not, i.e. it merges each propagation map generated for all scenarios in a global probabilistic map. It is important to notice that this method uses the same error function (equation 1) as the GA used.

As we have mentioned, $SAPIFE^3$ merges the two above described prediction methods including their advantages and demising their drawbacks. In particular, $SAPIFE^3$ reduces the number of total scenarios from a number such as hundreds of thousands to several hundreds, by optimizing the set of scenarios through the use of a GA. The combination of several individuals improves the results of the GA in case of sudden changes. That is, when conditions change hardly, the best individual found by the genetic algorithm at Calibration stage could be a very bad one in the Prediction stage. Nonetheless, if we consider the whole population, some individuals referred as to bad individuals during the Calibration stage, may be useful in the Prediction stage.

In the following section, we shall introduce the data assimilation module for the proposed architecture.

3 Data Assimilation

The Data Collection System component - (see figure 2) is the responsible to gather all information regarding the fire's environment, such as weather, topography and

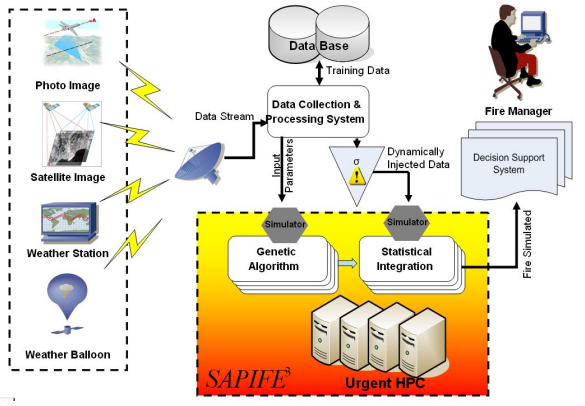


Fig. 2. Conceptual Design

terrain composition data (the combustible). This module must work alongside GIS (Geographical Information System) tools,i.e. MIRAMON [9]. This module must also be well connected to a network of weather stations such as the Network of Automatic Weather Stations of Catalonia's government (XEMA - Xarxa d'Estacions Meterologics Automatiques in catalan).

This module also injects data in real-time. Data is read from the weather stations through *ftp* connections, and then copied to a file inside the execution environment. The process responsible for the statistic integration monitors this file, and in case of changes on it, the changes are introduced in the form of replacing the worst individual who came from the *GA*.

4 Experimental Study

In november 2008, southern California was hit by devastating fires. The extreme conditions of the Santa Ana's winds [11], combined with the environment's low humidity, created the ideal conditions for fires as, for instance, the one known as "Freeway Complex Fire", which destroyed around 850 houses, and burnt more than 40.000 acres. The losses due to this fire were about 16 million dollars [7].

In order to test our DDDAS forest-fire propagation prediction system, we performed a series of postmortem experiments based in the conditions of the Freeway Complex Fire. The main objective of these experiments were to demonstrate the benefits of DDDAS for forest fire prediction, specially when environmental conditions are quite dynamic showing suddenly changes in wind speed and wind direction. This way, we are demonstrating the importance of the DDDAS systems for forest fire prediction, and in what way they affect the fire simulators' output, when conditions are dynamic and changes are sudden.

Fig. 3. Freeway Complex Fire view using Google Earth and MODIS Hotspot

The Freeway Complex Fire happened between the cities of Corona, Chino Hills, Yorba Linda, Brea and Anaheim, in the state of California. In this region, there are several weather stations, property of the Weather Underground [14]. The one chosen to gather data for our experiments was the KCAYORBA4 weather station, located at latitude 33.88 and longitude -117.79, inside the area affected by the fire.

This station was chosen because it monitors humidity, wind speed and wind direction every five minutes. We also used the MODIS Hotspot detection system [10], which allows fire data to be visualized into Google Earth using KML languaje [8] , so it is possible to verify the situation of the fires. Figure 3 shows data for november 16, 2008 where it is possible to visualize the KCAYORBA4 weather station at the bottom of the image.

The data available for this fire is quite extensive, therefore, in order to have reasonable experimentation times, we cropped the data region into a one square kilometer plot, with a slope of five degrees. This selected area is marked in figure 3 with X. In the reported experiments we recreate the wind conditions according to the data gathered by the KCAYORBA4 weather station in november 16, 2008, between 4:00 and 5:20 a.m. During this time spam, relevant changes in enviromental conditions occurs in a small period of time what allows us to show how sudden changes can affect traditional fire simulators, such as FireLib [4]. We also show how to improve spread prediction results applying DDDAS methods.

The evolution of wind speed and wind direction for the selected time interval is shown in figure 4. As we can observe, the behavior patterns both for wind direction and wind speed are quite fluctuating denoting huge variabilities in five minutes intervals as, for instance, between minutes 10 and 15, where wind speed changes from 2.7mph to 10.1mph. The same effect can be observed in wind direction, which changes in almost every time interval. Those changes are impossible to be predicted and that is the reason why the real-time dynamic data injection could became a crucial point.

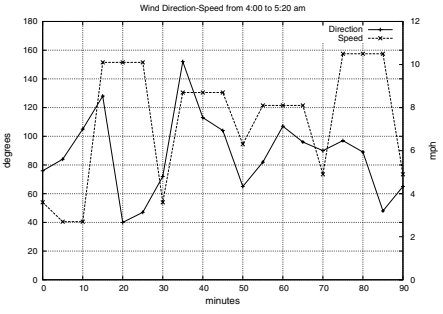


Fig. 4. Changes in wind speed and wind direction during one hour and twenty minutes

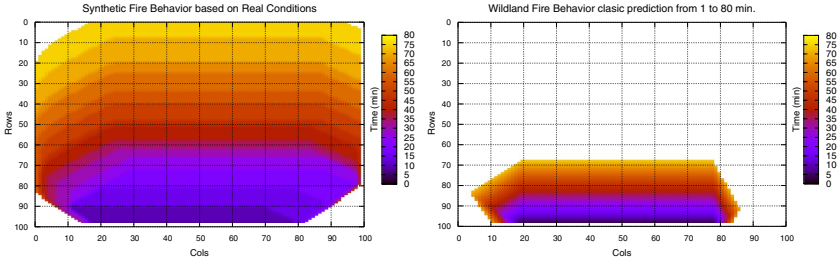


Fig. 5. Dimensions: 1000x1000 meters, 10m^2 cells, 5 slope, model 4(Mixed Chaparral)

As it was described in previous sections, the proposed DDDAS for forest fire behavior prediction needs to be fed with the map of the real observation of the fire propagation, for calibrating purposes. Since our experiments are dealing with a recreation of the real fire, we generate a synthetic real map propagation by manually varying wind speed and wind direction every 5 minutes following the KCAY-ORBA4 weather station data pattern. Consequently, we obtained a different propagation map every five minutes. All these maps were joined together, in one single propagation map, that goes from minute 0 to 80. We call this map the "synthetic propagation map based on real conditions" - see figure 5(a). However, in future work, this real map would be an aerial or satellite image of the fire's evolution.It is important to note that the only parameters that changed at each time interval were wind speed and wind direction. All other simulator parameters, such as vegetation model number 4 (Mixed Chaparral - typical from southern California [1]), slope and humidity, were kept constant during the whole simulation.

Figure 5(b) reproduces the propagation map generated by the simulator, when wind speed and wind direction are introduced at time zero and kept constant throughout the simulation (from minute 0 to minute 80). This case shows the prediction results provided by the classical prediction method where a single input parameters measurement are used for the complete simulation process.

Comparing this propagation map to the "synthetic propagation map based on real conditions" we can state the bad prediction quality provided by the classical prediction scheme because of the lack of considering dynamic conditions. In particular, the prediction error rate obtained in this case is more than 90% what is clearly unuceptable.

4.1 Experimental Results

As it was described in section 2, SAPIFE^3 is composed by the Genetic Algorithm (GA) and the statistical scheme S^2F^2M. In the proposed experiment, the real-time data injection is done after the GA stage and just in the beginning of S^2F^2M. This modification of the basic SAPIFE^3 has been called SAPIFE$^3_{rt}$. The particular GA's configuration for the reported experiments is: population size 500, generation 5, elitism 20, crossover probability 0.2 and mutation probability 0.01. The experimental results shown in this section, include the prediction results provided by those three dynamic data driven schemes.

Slope and vegetation model are assumed to be known, therefore they are set up as constants inputs for all experiments and schemes. As it has been previously described, the measurement of wind speed and wind direction are available, not only at the very beginning of the fire, but also every 5 minutes (recorded by KCAYORBA4 weather station). Although these data availability, the only dynamic data driven prediction scheme that can take advantage of such information is SAPIFE$^3_{rt}$ because of its ability to receive real-time data at execution time.

Since all compared methods work in a two stage scheme, we shown in two different graphics the results provided at the first stage (calibration stage, figure 6) and after the second stage where the whole prediction process has been finished (prediction stage figure 7). In figure 6, we can observe that GA is the scheme that provides better error adjustments, that means that it can find a set of parameters (individual) capable of reproducing the fire's propagation in a very similar fashion of the real fire for the time interval used for calibration.

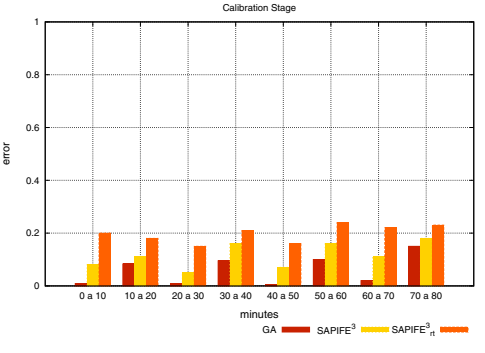


Fig. 6. Comparison between three methods at Calibration Stage

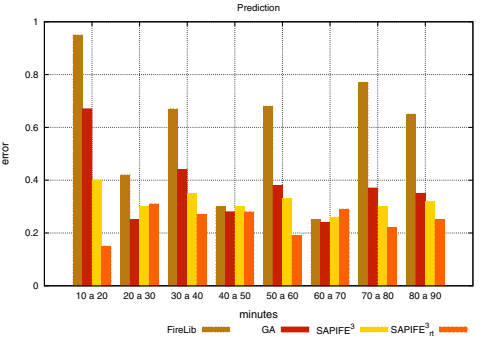


Fig. 7. Comparison between three methods in Prediction Stage

SAPIFE^3 also presents good adjustment ratios, however with slightly more error ratios than GA. It happens because when integrating each individual, some of them may not be adequate. Nevertheless, the error ratio is still low. The third column shows SAPIFE$^3_{rt}$ which is the method that denotes higher error rates at the calibration stage. That's happens because SAPIFE$^3_{rt}$ not only injects data at the prediction stage but also applies the same behavior at the calibration stage and, therefore, the data injected during the current calibration stage does not correspond to the time interval effectively used for the calibration stage masking the results. That means that to perform the calibration stage for time interval 0-10 minutes, the system is injecting data measured at a time posterior to minute 10. Although it looks like a disadvantage, when sudden change occurs it can be a very useful characteristic as it will be reported subsequently.

Figure 7 depicts the prediction error provided by each method once the prediction stage has finished. In this comparison we included the FireLib results representing the classical prediction method. As FireLib has no calibration stage, it was not included in the previous figure. It is important to notice that, although we are depicting time intervals that exactly last 10 minutes, in fact, the prediction results are provided before reaching the end of the corresponding interval time. However, we can not evaluate the goodness of the obtained prediction until reaching the end of the underlying time interval. That is the reason we plot as prediction interval the exact times. For example, in a time instant previous to t_{20}, the system will deliver the prediction fire behavior for time t_{20}, however, the real prediction validation will be performed only when fire propagation will reach time instant t_{20}. The same happen at each prediction step as shown in figure 7.

An immediate conclusion obtained from observing figures 7 is that FireLib prediction results are for all time intervals the worst. This fact states that the classical prediction scheme where no dynamic data driven approach is included, is a clear drawback of such a scheme. Furthermore, and taking into account the result discussed from figure 6, one can see that there is no a direct correlation between the results obtained in the calibration stage and the results provided by

the prediction stage, in fact, they apparently tend to have an inverse relation. For example, from minute 10 to 20, GA denotes a high error ratio, although at the calibration stage (from minute 0 to 10) it provided the best error adjustment. The same behavior also appears in most of the interval times. GA has an intrinsic drawback related to its impossibility of being aware of drastic changes in environmental conditions from calibration stage to the prediction stage. This penalty is more incident for the case studied bacause of the wind variability pattern used. Therefore, we can affirm that in the presence of sudden changes, the conditions in the moment of the calibration stage does not determine the prediction stage specially when either a classical approach or the GA scheme are applied. $SAPIFE^3$ and $SAPIFE^3_{rt}$ denote the the best prediction results and, in particular, $SAPIFE^3_{rt}$ is shown to be the best. The ability of injecting real-time data allows, for the case studied, to keep bounded the error ratio below 20% although in presence of drastic wind changes.

If we observe wind behavior in figure 4, we can see that it suffers from extreme change on its speed in minute 15. This change is taken into account by $SAPIFE^3_{rt}$ when performing the prediction stage. This fact represents a big advantage, because of this change will generate an increase in fire spread velocity that will not be considered otherwise. Consequentely, $SAPIFE^3_{rt}$ gets almost 50% less error than $SAPIFE^3$, who doesn't have any runtime data insertion mechanism - and who is going to notice the changes only in the range from 20 to 30 minutes. Besides, we can see that the improvement over GA is almost 70%, and more than 90% over FireLib.

In the time frame between 20 and 30 minutes, GA is the one who better performs. This happened because the wind conditions keeps quite similar between the adjustment and prediction phases. This turns the individual found in the 10 to 20 minutes time frame to be very good also for the next period. However, $SAPIFE^3$ and $SAPIFE^3_{rt}$ are very close to it, even in those stable conditions, which keep quite constant until minute 35, when again, they change a lot. This affects seriously the prediction of all methods except for $SAPIFE^3_{rt}$, which is able to get results with error ratios less than 30%.

5 Conclusions and Future Work

In this work, we presented a DDDAS for forest fire spread prediction with real time data injection. We performed a series of experiments based on the behavior of two most variable parameters: wind speed and wind direction. The data used to set up those experiments has been gathered from southern California fires.

The experiment results obtained shown that runtime data insertion improve prediction when conditions change suddenly during a fire. However, this dynamic data insertion must be performed only in the presence of sudden changes, to not disturb simulation results. This will be taken into account for future $SAPIFE^3$ versions, where it will be able to detect sudden changes automatically, and it will be able to decide whether data is going to be inserted or

not. This work also demonstrate that a conventional simulator can easily being ported to the proposed DDDAS system having a considerable improvement in its prediction quality. For this reason, we are developing a a general DDDAS framework for any kind of simulator on High Performance Computing (HPC) platforms. In order to introduce the Urgent factor into the systems (Urgent-HPC) we will use SPRUCE (Special PRiority and Urgent Computing Environment)[2] as a authorization system for allocation urgent sessions. This approach will provide new challenges such as dynamic data injection in grid environments.

References

1. Anderson, H.E., Forest, I., Station, R.E., Ogden, U.: Aids to Determining Fuel Models for Estimating Fire Behavior, tech. report INT-122, Agriculture Dept. Intermountain Forest and Range Experiment Station, U.S. Forest Service (1982)
2. Beckman, P., Nadella, S., Trebon, N., Beschastnikh, I.: SPRUCE A System for Supporting Urgent High-Performance Computing. In: Proc. IFIP International Federation for Information Processing, pp. 295–311. Springer, Boston (2006)
3. Bianchini, G., Cortés, A., Margalef, T., Luque, E.: Improved Prediction Methods for Wildfires Using High Performance Computing: A Comparison. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3991, pp. 539–546. Springer, Heidelberg (2006)
4. Collins, D.B.: FireLib User Manual & Tecnhical Reference (November 2006), `http://www.fire.org/`
5. Douglas, C.: Dynamic Data Driven Application Systems homepage (January 2008), `http://www.dddas.org`
6. Denham, M., Cortés, A., Margalef, T., Luque, E.: Applying a Dynamic Data Driven Genetic Algorithm to Improve Forest Fire Spread Prediction. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 36–45. Springer, Heidelberg (2008)
7. Orange County Fire Authority - OCFA homepage: Freeway Complex Fire Premliminary Report (February 2008), `http://www.ci.yorba-linda.ca.us/Fire_Prelim_Report.pdf`
8. KML Support homepage: KML Tutorial (November 2008), `http://code.google.com/apis/kml/documentation/mapsSupport.html`
9. MIRAMON Geographic Information System and Remote Sensing software homepage: What is MiraMon? (January 2008), `http://www.creaf.uab.es/MiraMon/what_mm/ENG/index.htm`
10. Justice, C.O., Giglio, L., Korontzi, S., Owens, J., Morisette, J.T., Roy, D., Descloitres, J., Alleaume, S., Petitcolin, F., Kaufman, Y.: The MODIS fire products. Remote Sensing of Environment 83, 244–262 (2002)
11. Raphel, M.N.: The Santa Ana Winds of California. Journal Earth Interactions 7(8), 1 (2003)

12. Rodriguez, R., Cortés, A., Margalef, T., Luque, E.: An Adaptive System for Forest Fire Behavior Prediction. In: Proc. 11th IEEE Int'l Conf. Conference on Computational Science and Engineering (CSE 2008), pp. 275–282. IEEE Computer Society, Washington (2008)
13. Rothermel, R.C.: How to predict the spread and intensity of forest and range fires, tech. report INT 143, Agriculture Dept. Intermountain Forest and Range Experiment Station Ogden, U.S. Forest service (1983)
14. Weather Underground homepage: History for KCAYORBA4 (November 2008), http://www.wunderground.com/weatherstation/

Event Correlations in Sensor Networks

Ping Ni, Li Wan, and Yang Cai

CIC Building, 4720 Forbes Ave Pittsburgh, Carnegie Mellon University
{pingni,liwan,ycai}@andrew.cmu.edu

Abstract. In this paper we present a novel method to mine the correlations of events in sensor networks to extract correlation patterns of sensors' behaviors by using an unsupervised algorithm based on a hash table. The goal is to discover anomalous events in a large sensor network where its structure is unknown. Our algorithm enables users to select the correlation confidence level and only display the significant event correlations. Our experiment results show that it can discover significant event correlations in both continuous and discrete signals from heterogeneous sensor networks. The applications include smart building design and large network data mining.

Keywords: Correlation, Visualization, Sensor Network.

1 Introduction

Discovering salient correlations between events in a large sensor network is valuable for reducing the number of sensors, unnecessary communication and energy consumption. For example, if we know the relations between temperature and humidity we can predict humidity through temperature. Sensors corresponding to humidity can then be removed from a smart building.

Kun-Chent [7] proposes a smart control algorithm to naturally adjust the thermal quality of the environment according to the interior and exterior environmental factors and the behavior of the inhabitants. They analyze correlations between sensors that are known in advance. In Kay's system [10] a user can pose a query to the system using a declarative language. Such a query defines the local events of interest and additional constraints on the sought for frequently occurring event patterns. In this system the frequent patterns are discovered only through some sensors so the calculation resources are constrained. What's more, it only explored the specific patterns of the user's query and did not display relations between sensors automatically. It can't predict events which will emerge in following time interval. Kay [14] focuses on embedded system pattern discovery which characterize the spatial and temporal correlations between events. It only defined the support parameter which is really not enough to discover correlations between events.

Here we address correlations in sensor networks by using an event-driven model for improving efficiency and effectiveness. We extract valuable patterns which are representative patterns of closed patterns [17] instead of complete patterns. Our contributions include: 1) discovering the significant patterns without necessary user specified support definitions, 2) prioritizing event correlations instead of complete correlations, and 3) visualizing the key event correlations in a network diagram.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 500–509, 2009.

2 Related Work

Mining for correlations [5] is recognized as an extremely important data mining task for its many advantages over data mining using association rules. Instead of discovering co-occurrence patterns in data as does association rule mining, correlation mining identifies the underlying dependency from the data set itself. Those infrequent but significant patterns that are too expensive to be revealed by association rule mining can now be discovered using correlation mining techniques. Zhang [3] discussed how to find two correlation sets. It can extract correlation between multiple series through a variable time window. But obviously it can't find a dynamic correlation. For example we might know that sometimes the motion of people can lead to the light being turned on. If in a short time there is a high dB sound present, the light can be also turned on. This is a burst event. Zhang's approach [3] couldn't find the correlations because there is a very short time correlation between the acoustic event and light being turned on. Indeed, acoustics and light have strong correlation too. Mattew [1] addresses the problem of online detection of unanticipated modes of mechanical failure given a small set of time series under normal conditions, with the requirement that the anomaly detection model be manually verifiable and modifiable. Ke [4] described the relations between the association rule and correlation. The correlation can be obtained when the parameter "support" is ignored by the defined association rule. It will introduce complex time questions and dramatically increases the memory demand if there are lots of items in data set when ignoring the support parameter. Sometimes the value of each attribute is not only of a Boolean type.

How to get exact patterns among attributes of a non-Boolean type presents a scientific challenge. Edith Cohen [6] found interesting associations without support pruning efficiently by using a compressing transformation to get an estimation matrix and then verified the validation of the transformation. Indeed it can find correlations between two columns which hold sparse data efficiently. But it is only for Boolean-type and only for correlations of two columns. With real-world data we need to extract relations between multiple columns like our sensor data set.

3 Problem Definition and Algorithm

In this study, there are over 200 sensors; here we only choose 6 of them . They are acoustic, light, motion, temperature, CO2, humidity. The sensors' names are abbreviated by their respective first initials (Acoustic=a, Light=l, Motion=m, Temp=t, Co2=c and Humidity=h).

a. Decreasing Memory Usage

We in fact needn't load the entire data set into the memory because the data set contains sparse data. We will maintain as static the rows which have concurrence events. Since we know that it is a very sparse data set with strong relations among the sensor data set, the emerging event states should have some relations that should be revealed according to common sense. The entire data space could also be compressed using common sense. We use the following data structure to compress our entire data set in a hash table.

Table 1. Hashtable structure

Key (event string)	Value (frequency)
Key1	F1
Key2	F2

From Table 1 we see that the same event string can be assigned the same key and their frequency would be incremented. The whole data set can be compressed like a FP-Growth [16] algorithm in which it compresses data set through FP-tree structure. This is verified using statistics from our entire data sets. Thirty nine patterns including various events took place in 83979 records of six sensors in two months from a real data set. Thirty nine patterns would be loaded into the memory for confidence calculation according to the event sets relation. The satisfied confidence between emerging events set is stored. So it needs only few memory spaces to hold all valuable patterns and we can work only on the compressed data set in hash table.

b. Problem Definition

Here we assumed we have events sets $E=\{E_1,E_2,...E_k....,E_n\}$, E is the event sets of this data set. E_k represent the k^{th} events set in E. The events are stored in a hash table as stated earlier. Any pairs in E are not equal. $E_k=\{e_{k1},e_{k2},...e_{ki}....,e_{kj}\}$, e_{ki} represents if *i* takes place e_{ki} would be set corresponding discrete value.

Here we assume $E_1=\{a\}, E_2=\{a,c\}, E_3=\{c,t\}, E_4=\{l,m\}$. For pair events we will have followed relations.

- **Full-Contained-relation:** all emerging events in an event set are all contained in another event set. For example $E_1 \subset E_2$, $\{a\}$ is contained in $\{a,c\}$.
- **Disjoint relations:** any emerging event in an event set is not contained in another event set. For example E_2 and E_4 ..
- **Part-containing-relation:** At least one but not all events which took place in one set are contained in another event set which hold at least one different event with previous event set. For example E_2 and E_3.

Here we will illustrate relations between valuable pattern sets and complete pattern sets. Traditionally we would explore whole complete combinations among the events space like $E_1 \to E_iE_k, E_1 \to E_iE_kE_j, E_1 \to E_i$. E_iE_k means the union of the events. Here we examine our questions from $E_1 \to E_iE_k$ and judge if E_iE_k is valuable. Here it would have followed cases. The following cases are described.

1. E_iE_k is contained in E, it means that there is an event set equal with E_iE_k, we assume $E_l = E_iE_k$. For $E_1 \to E_iE_k$ we can get from $E_1 \to E_l$ instead of E_iE_k.
2. E_iE_k is not contained in E, and not the subset of any element in E. so the $|E_iE_k|$ should be zero. So the $E_1 \to E_iE_k$ is zero.
3. E_iE_k is not contained in E, but it is the subset of one or more elements in E. for $E_1 \to E_iE_k$ is not valuable than $E_1 \to \sup erset(E_iE_k)$. Because E_iE_k is the subset of some event set (assumed E_a) in E. So we can get valuable patterns from $E_1 \to E_a$

For $E_1 \to E_i...E_k...E_j$ we can get similar reasoning recursively.

Obtain the entire valuable pattern set we only explore the correlation of the follows matrix.

$$\begin{array}{c} \\ E_1 \\ E_2 \\ \cdots \\ E_n \end{array} \begin{array}{c} \begin{array}{ccc} E_1 & E_2 & \cdots\cdots \quad E_n \end{array} \\ \begin{pmatrix} & E_1 \to E_2 & \\ E_2 \to E_1 & & \\ & & \\ & E_n \to E_2 & \end{pmatrix} \end{array} \tag{1}$$

For each pair of event sets in matrix (1) we will define an operator between them as follows.

1. Full-Contained-relation Confidence: like $\{a\}$ and $\{a,c\}$ we would get Confidence

$$Confidence(\{a\} \to \{a,c\}) = \frac{|ac|}{|a|} \tag{2}$$

2. Part-containing-relation Confidence: like {ac}->{ct}, so we get the confidence

$$Confidence(\{a,c\} \to \{c,t\}) = \frac{|c \to a|}{|c|} \tag{3}$$

3. Disjoint relation Confidence: like {c,t} and {m,l}, we get the confidence

$$Confidence(\{c,t\} \to \{m,l\}) = \frac{|ctml|}{|ct|} \tag{4}$$

After finishing the definitions of each relation, we give our prune definition for decreasing time cost.

We propose that if events $\{a,c\}$ has weak correlation with events set $\{c,t\}$, then $\{a,c\}$ will have weak correlation with all superset of events $\{a,c,t\}$ and prove it as follows:

If events $\{a,c\}$ occurs $k1$ times in whole data set, $\{c,t\}$ occurs $k2$ times. According to confidence definition of part containing relation,

$\frac{|ac \to ct|}{|ac|} = \frac{|act|}{|ac|} = \frac{k1}{k2} < \varepsilon$, ε represents the minimum confidence. Any superset of $\{a,c,t\}$ event would take place $k3$ times which are less than $k1$. So the confidence between $\{a,c\}$ and superset of $\{a,c,t\}$ would be less than ε.

Events which can't meet the minimum confidence will lead to pruning of their superset. We can summarize our algorithm below as algorithm 1 which incorporates a data preprocess algorithm from [11].

Algorithm 1

```
Input: min_confidence, cycle, time_slot, support.
//cycle, time_slot, support is referred from [11]
 Output: correlation between sensors.
1.  data preprocess
2.  Extract all co-occur events in the data set and add to Hashtable
    which key is the event set, and value is the emergence frequencies
    of combination events.
3.  For each elements in hash table{
4.  For each elements in hash table{
5.  Judge events relations according to above relation definition, we
    acquire events set A.and B.
6.  Getconfidence(A,B).
7.  If( Confidence(A → B) ≤ min_ confidence )
8.  According to pruning rule, the superset of A ∪ B would be pruned.
```

4 Performance Study

In this section, we evaluate the performance of the proposed algorithm and report the results that we have obtained using a real-world data set.

a. Experimental Results

First, we evaluate our algorithm 1. In algorithm 1 we set the cycle=5 and time_slot=5 minutes and support=0.1, support =0.3, support =0.5, support =0.9, support =1 respectively. Let X axis represents the patterns turn. Y axis represents the confidence between two event sets. See Figure 1.

From the comparison of figure in Fig.1, we can easily find that our sensors indeed have strong correlations as the support increases. When support=0.9 and 1 the relations are very obvious between sensors. From figure 1.e and 1.f we find that our sensors have strong correlations and have weak correlation. They have obvious group features between sensors but figure 1.a and figure 1.b are not obvious in displaying group features because support is too low. We can get effective results through our algorithm 1. From figure 1.e and 1.f we can find clustered information in which sensors can be considered as a strong correlation group. We cannot get the relation between t and h in either an *apriori* algorithm or FP-Growth algorithm when we set support=0.1.

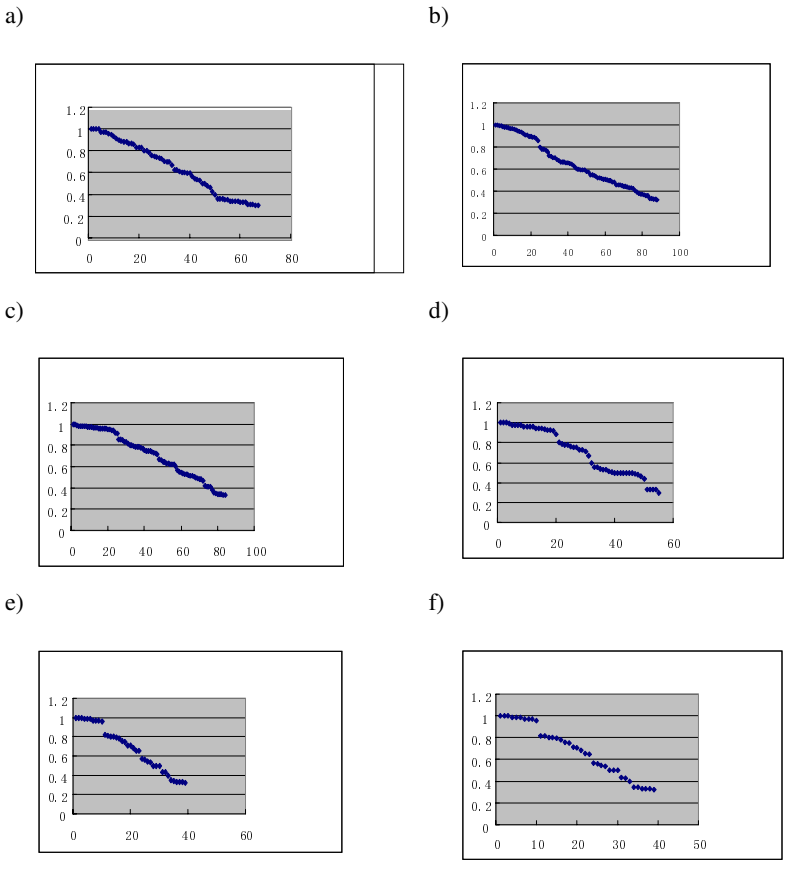


Fig. 1. a) support=0, b) support=0.1, c) support=0.3, d) support=0.5, e) support=0.9, f)support=1

The performance of our algorithm is tested using a laptop in which the CPU processor is 1.50 GHZ, memory is 760M. We split our real dataset into 6 parts (A,2),(B,5),(C,8),(D,10),(E,13), (X,Y) represents size of data set X is Y megabyte. Figure 1 gives the running time of algorithm 1 when confidence =0.1, 0.3, 0.7. As the confidence threshold increases and data size increases, our pruning effect is much more obvious. Our algorithm performs almost linear in time except for the memory limitation.

We use algorithm 1 to exploit a public data set [15]. First we extracted this public data set based on table 2. We list discovered key patterns that are described in table 3.

We can find that variance has strong relations between each other in the same sensor. And we also get a stronger correlation than others between t and v which has been declared by [15].

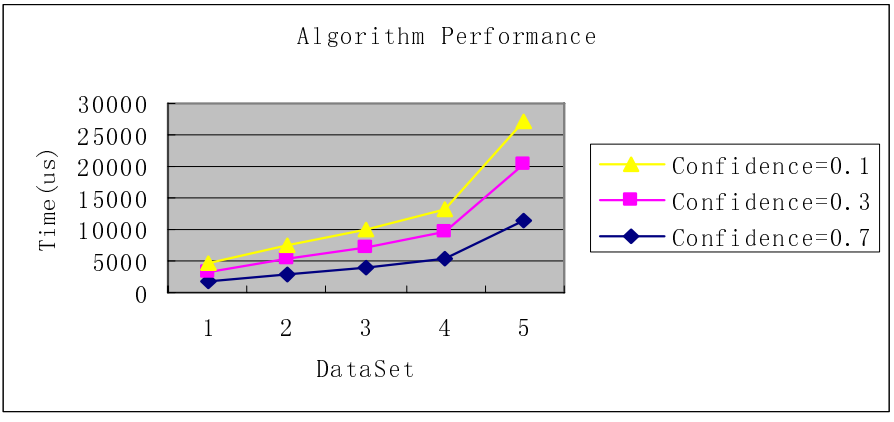


Fig. 2. Performance comparing

Table 2. MIT Data discrete standard

Sensor Name	Definition
temperature	Temp rise 10%, value is set 1, versa is 2, no change is 0.
humidity	Humidity rise 3% value is set 1, versa is 2, no change is 0.
Light	lux of light rise 10%, value is set 1, versa is 2, no change is 0.
voltage	Voltage rise 2% value is set 1, versa is 2, no change is 0.

Table 3. Correlation Patterns *Slot_num=5, c=5, support=1*

Patterns	Confidence	Discovered Patterns
t-v	0.76	t->h,v
t,v-h	0.66	v->t
t-h,v	0.51	l,v->t,h
h,v-t	0.36	t->v
l,v-t	0.33	v->h
l,v-h	0.33	
l,v-t,h	0.33	

5 Visualization of Event Correlations

The correlated sensor events can be visualized as a tree shape. Here we set our support parameter 1, confidence is 0.3, cycle is 5 and *time_slot=8.* We get the whole event set state diagram. From figure 3we found some noise existing in it according to common sense. Even though we employed a noise cleaning algorithm from [11] like $l, c \rightarrow m$, c is obviously an occasion event according to common sense since CO2 is a colorless gas which has no effect upon either lighting or motion sensors. We present algortihm2 to compress our patterns. We get our concise state diagram of Figure 4 through algorithm2 as follows.

Algorithm 2. Pattern Compress

```
Input: Correlated event sets
Output: Concised correlated patterns
1.  Find the nonzero in-degree node from state diagram.
2.  int patternCount[i];
3.  //table is two dimension array including each pattern in every rows
    //from state diagram. Like row 1, m,1. Row 2, m,l,h etc, the bit
    //will be set 1 in corresponding place.

4.  For each col i in table {    for each row j in table{
5.        patternCount[i] += table.get(j,i);
6.
7.  }
8.  if(patternCount[i]/allRows < concise_parameter)
9.         remove column i from table;
10. }
11. Return table;
```

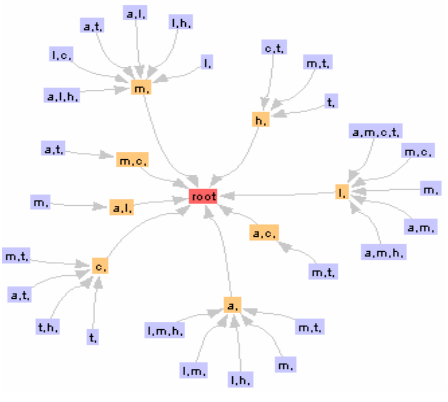


Fig. 3. Event Correlation diagram between sensors

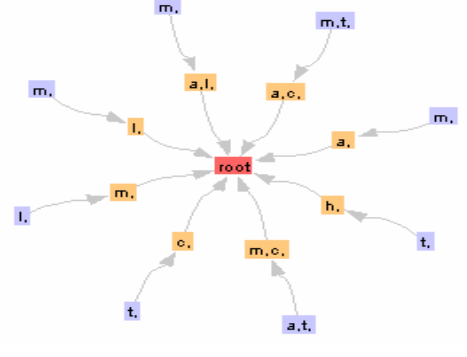


Fig. 4. Trimmed Event Correlation diagram with *concise_parameter=0.7*

Compared the tree structures between the images in Fig. 3 and 4, the state relation is suppressed and present more meaning relations in our sensor network according to commonsense.

6 Conclusions

We here present a novel method to extract correlations from a large number of sensors instead of using a traditional method based on an *apriori* algorithm and pattern growth[16] method. Our method is event-driven and discovers specific valuable patterns instead of a complete pattern set. We incoporate the algorithm from [11] within algorithm 1 in sensor networks to improve efficiency for discovering concise and accurately correlated patterns. We reclean noise from patterns and show concise and meaningful patterns through state diagram illustration. Our experiments verify that our novel method is both highly effective and efficient.

Acknowledgement

The authors would be to thank Brian Zeleznik for his help on reviews and revisions for this paper.

Reference

1. Mahoney, M.V., Chan, K.: Trajectory Boundary Modeling of Time Series for Anomaly Detection. SIGKDD Explorations Newsletter 7(2), 132–136 (2005)
2. Fabian, M.: Unsupervised Pattern Mining from Symbolic Temporal Data. SIGKDD Explorations Newsletter 9(1), 41–45 (2007)
3. Zhang, T., Yue, D., Gu, Y., Yu, G.: Boolean Representation Based Data-Adaptive Correlation Analysis over Time Series Streams. In: CIKM 2007 Information and knowledge management, pp. 203–212. ACM, New York (2007)
4. Ke, Y., Cheng, J.: Correlated Pattern Minging in Quantitative Databases. ACM Transactions on Database Systems 3(33), Article 14 (August 2008)
5. Ke, Y., Cheng, J.: Correlation search in graph databases. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 390–399. ACM, New York (2007)
6. Cohen, E., Datar, M.: Finding Interesting Associations without Support Pruning. IEEE Transaction on Knowledge and Data Engineering 1(13) (February 2001)
7. Tsai, K.-C., Sung, J.-T.: An Environment Sensor Fusion Application on Smart Building Skins. In: IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, June 2008, pp. 291–295 (2008)
8. Sharples, S., Callaghan, V., Clarke, G.: A Multi-agent Architecture for Intelligent Building Sensing and Control. International Sensor Review Journal, 1–8 (May 1999)
9. Hagras, H., Callaghan, V., Colley, M., Clarke, G.: A Hierarchical Fuzzy–genetic Multi-agent Architecture for Intelligent Buildings Online Learning, Adaptation and Control. Information Sciences 150, 33–57 (2003)
10. Kay, R.: Discovery of Frequent Distributed Event patterns in Sensor Networks. In: Verdone, R. (ed.) EWSN 2008. LNCS, vol. 4913, pp. 106–124. Springer, Heidelberg (2008)

11. Boukerche, A., Samarah, S.: A Novel Algorithm for Mining Association Rules in Wireless Ad Hoc Sensor Networks. IEEE Transactions on Parallel and Distributed Systems 19(7) (July 2008)
12. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rule. In: Proc. 20th Int'l Conf. Very Large Data Bases (VLDB 1994), pp. 487–499. Morgan Kaufrmann, San Francisco (1994)
13. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proceedings of the 11th international Conference on Data Engineering, pp. 3–14. IEEE Press, Los Alamitos (1995)
14. Cong, S., Han, J.: Parallel mining of closed sequential patterns. In: KDD 2005: Eleventh ACM SIGKDD international conference on knowledge discovery in data mining, Chicago, Illinois, pp. 562–567 (2005)
15. Intel Lab Data (2007), http://berkeley.intel-research.net/labdata/
16. Han, J., Pei, J., Yin, Y.: Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. In: Data Mining and Knowledge Discovery, 2000 ACM SIGMOD international conference on Management of data, vol. 8(1), pp. 1–12 (2000)
17. Zaki, M., Hsiao, C.: Charm: An Efficient Algorithm for Closed Itemset Mining. In: SDM 2002 (April 2002)

Chairs' Introduction to Workshop on Computational Finance and Business Intelligence

Yong Shi[1,2], Shouyang Wang[3], and Xiaotie Deng[4]

[1] Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing, 100190, China
yshi@gucas.ac.cn
[2] College of Information Science & Technology, University of Nebraska at Omaha, Omaha, NE 68182, USA
yshi@unomaha.edu
[3] Laboratory of Management, Decision and Information Systems, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, 100190, China
sywang@amss.ac.cn
[4] Department of Computer Science, City University of Hong Kong, China

Abstract. We have been organizing the Workshop on Computational Finance and Business Intelligence (CFBI) at International Conference on Computational Science (ICCS) since 2003. This workshop at ICCS, Baton Rouge, Louisiana, U.S.A., May 25-27, 2009 focuses on computational science aspects of asset and derivatives pricing, financial risk management, and related topics to business intelligence. It will include but not limited to modeling, numeric computation, algorithmic and complexity issues in arbitrage, asset pricing, future and option pricing, risk management, credit assessment, interest rate determination, insurance, foreign exchange rate forecasting, online auction, cooperative game theory, general equilibrium, information pricing, network band witch pricing, rational expectation, repeated games, etc.

1 Introduction

The 15 papers are accepted for CFBI, ICCS 2009. The first paper, "Lag-Dependent Regularization for MLPs applied to Financial Time Series Forecasting Tasks", by A. Skabar, proposes a lag-dependent regularization technique by which the influence that a lag has in determining the forecast value decreases exponentially with the lag. The second paper, "Bias-Variance Analysis for Ensembling Regularized Multiple Criteria Linear Programming Models", by P. Zhang, X. Q. Zhu, Y. Shi, explores bias-variance decomposition on RMCLP method, and concluded that boosting based RMCLP will mostly further improve the RMCLP models. The third paper, "Knowledge-rich Data Mining in Financial Risk Detection", by Y. Peng, G. Kou, Y. Shi, studies the concept of chance discovery into financial risk detection to build the knowledge-rich data mining process and therefore increase the usefulness of data mining results in financial risk detection. The fourth paper, "Smoothing Newton Method for l1 Soft Margin Data Classification Problem", by W. B. Chen, H. X. Yin,

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 513–514, 2009.

Y. J. Tian, presents a smoothing Newton method for solving the dual of the l1 soft margin data classification problem. The fifth paper, "Short-term Capital Flows in China : Trend, Determinants and Policy Implications", by H. Z. Yang, Y. P. Zhao, Y. J. Ze, builds a structural model-VECM to explore the determinants of net flows of short-term capital in China. The sixth paper, "Finding the Hidden Pattern of Credit Card Holder's Churn: a Case of China", by G.L. Nie, G. X. Wang, P. Zhang, Y. J. Tian, Y. Shi, provides a framework of the whole process of churn prediction of credit card holder. The seventh paper, "Nearest Neighbor Convex Hull Classification Method for Face Recognition", by X.F. Zhou, Y. Shi, introduces a novel classifier called Nearest Neighbor Convex Hull Classifier for face recognition. The eighth paper, "The Measurement of Distinguishing Ability of Classification in Data Mining Model and Its Statistical Significance", by L. L. Zhang, Q. X. Wang, J. Wei, X .Wang, Y. Shi, discusses the overlapping degree, and use K-S statistics to examine the confidence level of the results from data mining model, and construct the nonparametric statistics of AUC. The ninth paper, "Maximum Expected Utility of Markovian Predicted Wealth", by E. Angelelli, S. O. Lozza, proposes an ex-post comparison of portfolio selection strategies based on the assumption that the portfolio returns evolve as Markov processes. The tenth paper, "Continuous Time Markov Chain Model of Asset Prices Distribution", by E. Valakevičius, introduces a continuous time Markov chain model for asset dynamics. The 11th paper, "Foreign Exchange Rates Forecasting with a C-Ascending Least Squares", by L. Yu, X. Zhang, S.Y. Wang, proposes a modified least squares support vector regression (LSSVR) model. The 12th paper, "Multiple Criteria Quadratic Programming for Financial Distress Prediction of the Listed Manufacturing Companies", by Y. Wang, P. Zhang, G. L. Nie, Y. Shi, applies the Multiple Criteria Quadratic Programming (MCQP) model to predict financial distress of the listed manufacturing companies. The 13th paper, "Kernel Based Regularized Multiple Criteria Linear Programming Model", by Y. H. Zhang, P. Zhang, Y. Shi, extends RMCLP into solving non-linear problems by kernel trick. The 14th paper, "Retail Exposures Credit Scoring Models for Chinese Commercial Banks", by Y. H. Yang, G. L. Nie, L. L. Zhang, designs the target system of individual credit scoring with individual housing loans data, and established an individual credit scoring model including testing. The 15th paper, "The Impact of Financial Crisis of 2007-2008 on Crude Oil Price" by X. Zhang, L. Yu, S. Y. Wang, proposes an EMD-based event analysis approach for better estimation of the impact of extreme events on crude oil price volatility.

Finally, the chairs would like thank all of the program committee members as well as the reviewers for their valuable comments on the submissions. Without their support, the workshop cannot continue as it is now. We will work hard to participant in ICCS 2010 again.

Lag-Dependent Regularization for MLPs Applied to Financial Time Series Forecasting Tasks

Andrew Skabar

Department of Computer Science and Computer Engineering
La Trobe University, Victoria 3086 Australia
a.skabar@latrobe.edu.au

Abstract. The application of multilayer perceptrons to forecasting the future value of some time series based on past (or *lagged*) values of the time series usually requires very careful selection of the number of lags to be used as inputs, and this must usually be determined empirically. This paper proposes a regularization technique by which the influence that a lag has in determining the forecast value decreases exponentially with the lag, and is consistent with the intuitive notion that recent values should have more influence than less recent values in predicting future values. This means that in principle an infinite number of dimensions could be used. Empirical results show that the regularization technique yields superior performance on out-of-sample data compared with approaches that use a fixed number of inputs without lag-dependent regularization.

Keywords: Multilayer perceptrons, financial time series forecasting, weight regularization.

1 Introduction

Over the last two or so decades there has been much interest in applying multilayer perceptron (MLP) models to financial time series forecasting tasks. The appeal of MLPs is their property of being universal function approximators; that is, they are able to approximate any target function to arbitrary degree of accuracy [1]. However, the ability of a model to achieve good performance on in-sample data does not guarantee that it will perform well in forecasting out-of-sample data, and this is particularly a concern in the case of models such as MLPs, which, due to their complexity, together with the high level of noise present in financial time series, can very easily overfit the training data. It is not surprising, then, that there has been considerable debate about whether non-linear models such as MLPs are able to provide any better performance financial time series forecasting than linear or random walk models [2–4].

The main factor determining the complexity of an MLP model (and hence its propensity to overfit training data) is the number of weights, and this depends on two factors: the number of inputs to the model, and the number of hidden layer units. Since each input fans out to each hidden unit, the total number of weights will be a function of the product $D \cdot h$, where D is the input dimensionality and h is the number of hidden units; hence, choosing an appropriate value for these two parameters plays

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 515–523, 2009.

an important role in determining the overall complexity of the model. Fortunately, other means also exist for mitigating against the effect of model complexity. For example, it is common to include a regularization term into the error function, the purpose of which is to impose a penalty against large magnitude weights, thereby controlling the effective complexity of the model [5].

The main problem with which we are concerned in this paper is how to select an appropriate number of inputs. As is common practice on financial time series forecasting tasks, we assume that the input to the MLP is a vector of delayed returns $(r_{t-1}, r_{t-2}, \ldots, r_{t-D})^T$, where r_{t-n} is the return n days prior to day t (and which we will refer to as the n^{th} 'lag'). The problem, then, is how to select an appropriate value for D. If D is too small, then we may miss out on detecting important patterns in the time series; if D is too large then we run the risk of modeling noise in the training data, thereby leading to overfitting. A common practice in determining the optimum input dimensionality is to experiment with a range of values, and to then use some model selection criteria such as the Akaike Information Criterion (AIC) [6] or the Schwarz criterion (BIC) [7] to make the final selection [8]. Typically, the number of lags selected lies in the range two to five; however, there can be dramatic differences in performance observed by increasing the dimensionality by just one.

Motivated by the intuitive notion that recent values should have more influence than less recent values in predicting future values, in this paper we propose a weight regularization technique by which the influence that a lag has in determining the forecast value decreases exponentially with the lag. This means that in principle an infinite number of dimensions can be used. Empirical results show that the regularization technique yields superior performance on out-of-sample data compared with approaches that use a fixed number of inputs without lag-dependent regularization.

The remainder of this paper is structured as follows. In Section 2, we describe previous work, in which we used a generative model to predict the probability of an upward/downward direction of change in the value of a return series. Specifically, we show how the covariance matrix used to estimate densities can be parameterized in such a way that the influence of past returns decreases exponentially with time, thus allowing an effectively infinite input dimensionality. In Section 3 we show how this idea of exponentially decreasing influence of past returns can be implemented in discriminative models such as MLPs. Section 4 provides empirical results of applying the technique to several datasets, and Section 5 concludes the paper.

2 Related Work

In [9] we showed how the intuitive notion that recent values of the time series should to be more influential than less recent values could be implemented within a generative model. Assuming a set of examples $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ where $\mathbf{x}_n = (r_{n-1}, r_{n-2}, \ldots, r_{n-D})^T$ is a vector of delayed returns on day n, the objective was to predict the posterior probability of a positive return on day n. Since each $\mathbf{x}$ belongs to one of two classes, which we will denote as C_+ for positive returns and C_- for negative returns, the problem is thus a binary classification problem. Using a Parzen density estimation [10,11], the probability density function for examples belonging to C_+ can be estimated as

$$p(\mathbf{x} \mid C_+, \Sigma) = \frac{1}{|\mathbf{X}_+|} \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \sum_{n=1}^{|\mathbf{X}_+|} \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{x}_{+n})^T \Sigma^{-1} (\mathbf{x}-\mathbf{x}_{+n})\right) \quad , \tag{1}$$

where $\sum$ is the covariance matrix and $\mathbf{X}_+$ is the set of examples belonging to C_+. The density $p(\mathbf{x} \mid C_-, \Sigma)$ can be calculated similarly. These densities can then be combined using Bayes' Theorem to estimate the posterior probabilities $P(C_+ | \mathbf{x}, \Sigma)$ and $P(C_- | \mathbf{x}, \Sigma)$, which represent the probability respectively of a positive and negative return.

Since we expect that the importance of lags to decrease with the length of the lag, this suggests that the kernel used to estimate a density should not be symmetrical (i.e., spherical), but of a form such that its width along dimensions corresponding to recent returns is smaller than its width along dimensions corresponding to less recent returns. Assuming that significance decreases exponentially with the length of the lag, we proposed the following expression for the variance in the direction corresponding to the n^{th} lag:

$$v_{t-n} = a\, \mathrm{e}^{k(n-1)} \tag{2}$$

where v_{t-n} ($n \in \{1, 2, ..., D\}$) is the variance of the kernel in the direction parallel to the axis corresponding to lag r_{t-n}, k is an exponential scaling factor, and a is the variance parallel to the first lag. These values are thus the diagonal components of the covariance matrix, with all off-diagonal components being equal to zero.

In general, the higher the variance of the time series, the larger will be the value of a. The value of k will depend on how rapidly the influence of lags in predicting future values diminishes with the length of the lag. For example, a series in which only the most recent value is significant in predicting future values will have a large k value (i.e., rapid decay in significance, or, equivalently, rapid increase of variance in dimension parallel to that lag). Conversely, time series in which previous lags are significant in predicting future values will have a smaller k value.

3 MLPs for Time Series Forecasting

An MLP is a function of the following form:

$$f(\mathbf{x}^n) = h(u) \;\; where \;\; u = \sum_{j=0}^{Q} w_{kj}\, g\left(\sum_{i=0}^{P} w_{ji} x_i^n\right) \tag{3}$$

where P is the number of inputs, Q is the number of units in a hidden layer, x_i^n is the input at unit i from example n, w_{ji} is a numerical weight connecting input unit i with hidden unit j, and w_{kj} is the weight connecting hidden unit j with output unit k. The function $g(x)$ is either a sigmoid (i.e., $g(x) = (1 + \exp(-x))^{-1}$) or some other continuous, differentiable, nonlinear function. For regression problems $h(u)$ is the identity function (i.e., $h(u) = u$), and for classification problems $h(u)$ is a sigmoid.

Thus, an MLP with some given architecture and weight vector **w**, provides a mapping from an input vector **x** to a predicted output y given by y = f (**x**, **w**). For time series forecasting tasks performed in this paper, the vector **x** = (x_1, x_2, …, x_D) is assumed to be a vector of lagged returns in which $x_1 = r_{t-1}$, $x_2 = r_{t-2}$, etc. Given some data, D, consisting of n independent items ($\mathbf{x}^1$, y^1), …, ($\mathbf{x}^N$, y^N), the objective is to find a suitable **w**.

3.1 Maximum Likelihood Training

The conventional approach to finding the weight vector **w** is to use a gradient descent method to find a weight vector that minimizes the error between the network output value, f(**x**, **w**), and the target value, y. In the following we assume that the objective is to predict the direction of change in the next value of the return series; more specifically, the probability that the direction of change is positive. Therefore, under the assumption that the target values are binary (y = 1 for a positive change; y = 0 for a negative change), then the likelihood of observing the training data D, given some weight vector **w** is given by

$$\begin{aligned} p(D \mid \mathbf{w}) &= \prod_n \left(f(\mathbf{x}^n, \mathbf{w})^{y^n} + (1 - f(\mathbf{x}^n, \mathbf{w}))^{(1-y^n)} \right) \\ &= \exp \sum_n \left(y^n \ln f(\mathbf{x}^n, \mathbf{w}) + (1 - y^n) \ln(1 - f(\mathbf{x}^n, \mathbf{w})) \right) \end{aligned} \tag{4}$$

We wish also to specify a prior distribution for the weights, and we assume that this distribution is Gaussian with mean 0 and inverse variance α. Thus,

$$p(\mathbf{w}) = \left(\frac{\alpha}{2\pi} \right)^{1/2} \exp\left(-\frac{\alpha}{2} \sum_{i=1}^{m} w_i^2 \right) \tag{5}$$

We wish to find the weight vector which maximises the product of the likelihood and the prior. This can be shown to be equivalent to minimizing the following error term:

$$E = -\sum_{n=1}^{N} y^n \ln f(\mathbf{x}^n, \mathbf{w}) + (1 - y^n) \ln(1 - f(\mathbf{x}^n, \mathbf{w})) \; + \; \frac{\alpha}{2} \sum_{i=1}^{m} w_i^2 \tag{6}$$

The first term in this expression is commonly referred to as the ‘cross-entropy error’; the second term is a ‘regularization’ term. Effectively, the regularization term punishes weight vectors in which the magnitude of weights is large, thereby helping reduce the possibility of overfitting to the training data. Note that aside from the weights, there is only one free parameter in the above equation: α, which specifies the inverse variance for the weights prior.

3.2 Lag-Dependent Regularization

The regularization term in Equation 6 treats all weight equally. However, by separating weights into groupings, it is possible to implement a lag-dependent regularization, which achieves a similar result to the variance parameterization described in the case of generative models. Consider the use of non-spherical kernels of the form described in

Section 2 above. From a discriminative classifier perspective, this is effectively making the assumption that noise in the training examples varies in such a way that input dimensions corresponding to recent returns are less noisy than inputs corresponding to less recent returns. We can build this assumption into an MLP by using separate regularization coefficients for different families of input-to-output layer weights. Specifically, weights fanning our from inputs corresponding to recent returns should have smaller weight-regularization coefficients than weights fanning out from inputs corresponding to less recent returns.

We first consider weights in the input-to-hidden layer weights. We use the notation $w_{i_p h_q}$ to represent the weight connecting input unit i_p with hidden unit h_q (i_0 represents the bias). Separating these weights into D+1 groups, where D is the input dimensionality and Q is the number of units in the hidden layer, we have

$$\frac{\alpha}{2}\left(\sum_{j=1}^{Q} w_{i_0 h_j}{}^2 + \sum_{j=1}^{Q} w_{i_1 h_j}{}^2 + \sum_{j=1}^{Q} w_{i_2 h_j}{}^2 + \ldots + \sum_{j=1}^{Q} w_{i_D h_j}{}^2\right) \tag{7}$$

We now apply exponential scaling factor of $e^{k(n-1)}$ to weights fanning from the input nodes 1 to D giving

$$\frac{\alpha}{2}\left(\sum_{j=1}^{Q} w_{i_0 h_j}{}^2 + e^{0k}\sum_{j=1}^{Q} w_{i_1 h_j}{}^2 + e^{k}\sum_{j=1}^{Q} w_{i_2 h_j}{}^2 + \ldots + e^{(D-1)k}\sum_{j=1}^{Q} w_{i_D h_j}{}^2\right) \tag{8}$$

which can be more succinctly expressed as

$$\frac{\alpha}{2}\left(\sum_{j=1}^{Q} w_{i_0 h_j}{}^2 + \sum_{n=1}^{D}\left(e^{k(n-1)}\sum_{j=1}^{Q} w_{i_n h_j}{}^2\right)\right) \tag{9}$$

The hidden-to-output layer weights can be treated as a single group. Incorporating these, the full regularization expression becomes

$$\frac{\alpha}{2}\left(\sum_{j=1}^{Q} w_{i_0 h_j}{}^2 + \sum_{n=1}^{D}\left(e^{k(n-1)}\sum_{j=1}^{Q} w_{i_n h_j}{}^2\right) + \sum_{j=0}^{Q} w_{h_j o}{}^2\right) \tag{10}$$

Note that as with the generative approach described in Section 2, this modification still introduces two additional parameters: a parameter controlling the rate at which the value of the regularization coefficient varies with input lag k, and a parameter controlling the overall level of regularization, α.

4 Experiments

We have applied the approach we have described to the daily close price of three financial time series: the Australian All Ordinaries (AORD) index, the Dow Jones Industrial Average (DJIA) index, and the Australian–U.S. Foreign Exchange (AUSE) rate. In each case we used a 20-year out-of-sample forecast period from 1 January

1987 to 31 December 2006. Forecasts were performed in 25-day forecast windows, in which the model was constructed using the data points immediately preceding the forecast period. The number of data points used for model construction was 500 (approx. 2 years). This model was then used to predict the directional change probability for each data point in the forecast window. We note that the 25-day forecasting window period was chosen entirely for computational reasons, and there is no reason why a separate model cannot be constructed for a single prediction. The number of lags used in the input vector (i.e., the input dimensionality) was five. The number of hidden layer units was 50. As with the work described in [9], here, too, we are concerned with predicting the direction of change. That is, rather than forecasting the values of the return, we forecast the (probability of) the sign of the return.

4.1 Testing Directional Forecast Accuracy

An obvious measure of direction-of-change forecast accuracy is the fraction of days in some test period for which the sign is predicted correctly, and we refer to this as the *sign ratio* (*SR*). We would like to know whether the value of *SR* differs significantly from what we would expect if the signs of the actual and forecast predictions were independent. If P is the fraction of days in the out-of-sample test period for which the actual movement is up, and $1\,\hat{P}$ is the fraction of days for which the predicted movement is up, the expected fraction of days corresponding to a correct upward prediction is $P \times \hat{P}$, and the expected fraction of days corresponding to a correct downward prediction is $(1-P) \times (1-\hat{P})$. Thus, the expected fraction of correct predictions is $(P \times \hat{P}) + ((1-P) \times (1-\hat{P}))$. Since this is just the success ratio that we would expect if the signs of the actual and forecast predictions are independent, this is known as the *sign independence ratio* (*SRI*) [12]:

$$SRI = (P \times \hat{P}) + ((1-P) \times (1-\hat{P})), \tag{7}$$

Pesaran & Timmermann (1992) show how *SR* and *SRI* can be combined to produce a directional accuracy test in which values for the test are normally distributed under the assumption that predicted and forecast values are independently distributed. The variance in *SRI* and *SR* can be calculated as follows:

$$\text{var}(SRI) = \frac{1}{n}[(2\hat{P}-1)^2 P(1-P) + (2P-1)^2 \hat{P}(1-\hat{P}) + \frac{4}{m} P\hat{P}(1-P)(1-\hat{P})] \tag{8}$$

$$\text{var}(SR) = \frac{1}{n} SRI(1-SRI). \tag{9}$$

The value of the Pesaran-Timmermann test, which we refer to as the *PT-score*, is given by:

$$\text{PT-score} = \frac{SR - SRI}{\sqrt{\text{var}(SR) - \text{var}(SRI)}} \sim N(0,1), \tag{10}$$

Although the PT-score provides a measure of how statistically significant a set of forecasts is, it does not distinguish between models that differ in some important

respects. For example, it is often useful to analyze results of classification tasks using a confusion matrix. For binary classification tasks, this will be a 2 by 2 matrix

$$\begin{array}{c} \\ \text{Pred +} \\ \text{Pred –} \end{array} \begin{array}{c} \begin{array}{cc} \text{Act +} & \text{Act –} \end{array} \\ \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} \end{array}$$

where the rows represent predicted positives and predicted negatives, and the columns represent actual positives and actual negatives. The problem is that many different confusion matrices can result in the same PT-score. Thus, although the Pesaran-Timmermann test provides a measure of the statistical significance of a set of directional forecasts, it does not take into account the different nature of the models in respect to their ability to predicted positives/negatives.

One of the advantages of predicting the posterior probability of a directional change is that we can then use these probabilities to select the threshold for our classification. For example, we may only choose to make a directional forecast of *up* if the posterior probability exceeds a value of, say, 60%. For this reason, it is important to have a measure of how our classifier performs not at just a 50% decision threshold, but across the whole range of thresholds. Receiver Operating Characteristic (ROC) curves provide such a measure. Plotting the true positive rate against the false positive rate for decision thresholds from 0 to 1 produces an ROC curve. The area under the curve provides a convenient single-value summary of the classifier's performance.

4.2 Results

Table 1 and 2 show the respectively the PT-score and area under ROC curve for the Australian All Ordinaries Index corresponding to a range of values for α and k from Equation 10. The highest value in each table is shown in bold, and in both tables corresponds to $\alpha = 1.50$ and $k = 0.60$. Table 3 show the critical values for PT-score. Thus the observed PT-score of 5.135 is significant at the 10^{-6} level.

Table 1. PT-scores for Australian All Ordinaries (AORD) index

A	k									
	0.08	0.10	0.20	0.40	0.60	0.80	1.00	1.50	2.00	3.00
0.000	1.472	1.472	1.472	1.472	1.472	1.472	1.472	1.472	1.472	1.472
0.001	1.404	1.404	1.355	1.472	1.671	1.915	2.126	2.058	2.626	1.404
0.010	2.099	2.118	2.141	2.212	2.813	2.896	2.856	3.174	1.570	2.099
0.100	2.103	2.081	1.964	1.979	2.224	2.889	3.028	2.228	2.618	2.103
0.500	2.832	2.332	3.039	1.916	3.523	2.607	2.194	2.776	3.284	2.832
1.00	3.224	4.156	2.220	4.297	3.242	4.042	3.775	3.711	2.194	3.224
1.50	4.003	3.438	2.649	3.550	**5.135**	4.087	4.110	2.814	2.469	4.003
2.00	2.968	3.467	1.800	2.840	3.786	3.584	2.299	0.421	0.612	2.968
10.0	1.349	1.890	2.719	2.468	1.436	0.387	0.928	-1.266	1.496	1.349

Table 2. Area under ROC curve for Australian All Ordinaries (AORD) index

α	*k*									
	0.08	0.10	0.20	0.40	0.60	0.80	1.00	1.50	2.00	3.00
0.000	0.512	0.512	0.512	0.512	0.512	0.512	0.512	0.512	0.512	0.512
0.001	0.511	0.511	0.510	0.512	0.512	0.513	0.517	0.521	0.523	0.511
0.010	0.515	0.516	0.516	0.516	0.519	0.520	0.522	0.523	0.519	0.515
0.100	0.516	0.514	0.515	0.518	0.520	0.526	0.524	0.521	0.523	0.516
0.500	0.521	0.523	0.529	0.527	0.528	0.531	0.526	0.525	0.530	0.521
1.00	0.533	0.534	0.530	0.537	0.528	0.537	0.536	0.532	0.520	0.533
1.50	0.533	0.531	0.529	0.540	**0.545**	0.541	0.539	0.529	0.523	0.533
2.00	0.529	0.533	0.526	0.532	0.540	0.540	0.529	0.502	0.510	0.529
10.0	0.519	0.523	0.524	0.522	0.513	0.511	0.508	0.492	0.501	0.519

Table 3. Critical values for PT-score

Sig. level	0.05	0.01	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}
PT-score	1.645	2.327	3.090	3.719	4.265	4.753	5.199	5.612

By observing the column corresponding to $k = 0.6$, it can be seen from both tables that as α is increased the PT-score and Area steadily rise to a maximum value, and then begin to decline. This can be explained by the fact that a small value of α provides only a small penalty against large weights, thus leading to overfitting on the training data. Conversely, a large α imposes a high penalty on large weights, leading to underfitting on the training data. In both cases (i.e., overfitting and underfitting), we have suboptimal performance on holdout data.

Now consider the rows of each table. It can be seen that the maximum values in each row correspond to k values in the vicinity of 0.4 to 0.8. While the pattern observed when k is increased from 0.08 to 3.00 is not one of steadily rising to a maximum, and then declining (i.e., the maximum is not as clearly pronounced as is the case for varying α), the value of k does clearly affect the performance on holdout data. In fact, it is interesting to compare these results with those obtained by using a fixed number of inputs without any differentiation on the prior distribution of weight families. Using 1, 2, 3, 4 and 5 inputs yields respectively the following values for the area under ROC curve: 0.534, 0.523, 0.525, 0.534 and 0.529. These values are well below the value of 0.545 achieved using differentiated priors. It is interesting to note that the best values for the MLP approach (PT-score = 5.135, Area = 0.545) are very close to those for the density estimation-based approach described in Section 2 (PT-score = 5.704, Area = 0.543).

We performed the same experiments on the Australian–U.S. Foreign Exchange rate and the Dow Jones Industrial Average index. For the exchange rate, the best values obtained for PT-score and Area under ROC curve were 2.72 and 0.517, which are not as significant as the values of 3.22 and 0.521 obtained using the density estimation-based approach. Results for the DJIA were not statistically significant, either for the neural networks approach or the density estimation-based approach.

5 Conclusions

A lag-dependent regularization technique has been proposed for use with MLPs applied to financial time series forecasting tasks. The technique is motivated by the intuitive notion that recent values of the series should have more influence than less recent values in predicting future values. The technique has been tested on three financial datasets, with directional forecast accuracy found to be significant on two of these datasets. A disadvantage of the MLP approach, as compared with a density estimation-based approach that we have previously proposed, is that in addition to the value of α and k, MLPs also depend on factors such as starting weights. We are currently applying the regularization technique described here within a Bayesian MLP learning framework. The integrative nature of the Bayesian framework is expected to reduce some of the variation between results obtained from different starting weights.

References

1. Cybenko, G.: Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals and Systems 2, 304–314 (1989)
2. Adya, M., Collopy, F.: How effective are neural networks at forecasting and prediction? a review and evaluation. Journal of Forecasting 17, 481–495 (1998)
3. Chatfield, C.: Positive or negative? International Journal of Forecasting 11, 501–502 (1995)
4. Tkacz, G.: Neural network forecasting of Canadian gdp growth. International Journal of Forecasting 17, 57–69 (2001)
5. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (1995)
6. Akaike, H.: A new look at statistical model evaluation. IEEE Transactions on Automatic Control AC-19, 716–723 (1974)
7. Schwarz, G.: Estimating the dimension of a model. Annals of Statistics 6, 461–464 (1978)
8. Franses, P.H., van Dijk, D.: Non-Linear Time Series Models in Empirical Finance. Cambridge University Press, Cambridge (2000)
9. Skabar, A.: A kernel-based technique for direction-of-change financial time series forecasting. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part II. LNCS, vol. 5102, pp. 441–449. Springer, Heidelberg (2008)
10. Parzen, E.: On the estimation of a probability density function and mode. Annals of Mathematical Statistics 33, 1065–1076 (1962)
11. Duda, R.O., Hart, P.E.: Pattern Classification and Scene Analysis. John Wiley and Sons, New York (1974)
12. Pesaran, M.H., Timmermann, A.: A simple non-parametric test of predictive performance. Journal of Business & Economic Statistics 10, 461–465 (1992)

Bias-Variance Analysis for Ensembling Regularized Multiple Criteria Linear Programming Models*

Peng Zhang[1], Xingquan Zhu[2], and Yong Shi[1,3]

[1] FEDS Research Center, Chinese Academy of Sciences, Beijing, 100190, China
[2] Dep. of Computer Sci. & Eng., Florida Atlantic University, Boca Raton, FL, 33431, USA
[3] College of Inform. Science & Technology, Univ. of Nebraska at Omaha, Nebraska, USA
zhangpeng04@gmail.com, xzhu3@fau.edu, yshi@gucas.ac.cn

Abstract. Regularized Multiple Criteria Linear Programming (RMCLP) models have recently shown to be effective for data classification. While the models are becoming increasingly important for data mining community, very little work has been done in systematically investigating RMCLP models from common machine learners' perspectives. The missing of such theoretical components leaves important questions like whether RMCLP is a strong and stable learner unable to be answered in practice. In this paper, we carry out a systematic investigation on RMCLP by using a well-known statistical analysis approach, bias-variance decomposition. We decompose RMCLP's error into three parts: bias error, variance error and noise error. Our experiments and observations conclude that RMCLP'error mainly comes from its bias error, whereas its variance error remains relatively low. Our observation asserts that RMCLP is stable but not strong. Consequently, employing boosting based ensembling mechanism RMCLP will mostly further improve the RMCLP models to a large extent.

Keywords: Bias variance decomposition, RMCLP, Ensemble.

1 Introduction

Due to the strong application-driven nature, the fields of data mining and optimization are becoming heavily intermingled than ever [1]. Experts used to traditionally work in optimization have not gradually shifted to the data mining field, or vice versa. In 2001, Shi et al. [2] proposed a multiple criteria linear programming (MCLP) model for classification. The promising results of MCLP motivate widely applications in business intelligence. Most recently, based on MCLP, a much more sophisticated model called Regularized Multiple criteria Linear Programming (RMCLP) is proposed [3] and reported exciting results on some UCI benchmark datasets. On the other hand, empirical studies on some other UCI benchmark datasets also show significant drop of accuracy of RMLCP. These observations motivate us to step further to look into

* This research has been partially supported by a grant from National Natural Science Foundation of China (#90718042, #70621001, #70531040, #70501030, #10601064, #70472074, #60674109), National Natural Science Foundation of Beijing #9073020, 973 Project #2004CB720103, Ministry of Science and Technology, China and BHP Billiton Co., Australia.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 524–533, 2009.

the inherent characteristics of RMCLP in three aspects: (1) whether RMCLP is a strong classifier which is supposed to achieve low prediction errors on any testing sets; (2) whether RMCLP is a stable classifier with very little fluctuation of its performance; and (3) which ensemble method is eligible to enhance RMCLP's performance.

Bias-variance analysis is a powerful tool to study learning algorithms [4,5,6,7], mainly because that bias-variance decomposition can provide first hand evidence on the stability and the weakness of the classifiers built from the learning algorithms. To a specific learning algorithm $\mathcal{L}$, bias-variance analysis decomposes its error into three parts: bias error, variance error and noise error. Bias error comes from the inherent shortage of learners. For example, a linear learner will have a high bias error on a non-linear dataset. Variance error is related to the learner's stability on different samples. When the sample is changed, the learner probably will generate a different model which has different prediction accuracy. This shifting of performance with the changing of sample is called the variance error. Noise error is assumed to exist in every sample. For example, two examples may have the same attribute values while having different labels, and this discrepancy generates the noise error. A strong learner is supposed to have low bias error and variance error. For a given variance error, a strong learner is more likely to have a low bias error. On the other hand, a stable classifier is supposed to have low variance error, which means the learner fluctuates very little on different training samples. Besides applications in studying learning algorithms, bias variance decomposition also provides a rationale to develop ensemble methods [7]. If a learner is a weak learner (the opposite side of strong learner), boosting (a well-known ensemble mechanics) can be used to transform the weak learner into a strong learner by combing models trained on weighted training instances [8]. In contrast, if a learner is an unstable one, bagging (another well-known ensemble mechanics) can be used to transform it into a stable learner by equally combining each base learner together [9].

In this paper, to investigate why and how RMCLP works, we decompose RMCLP model's error by bias variance decomposition. Moreover, by observing where the error mainly comes from, we can develop an appropriate ensemble method to enhance RMCLP performance. For instance, if bias error accounts for the heaviest part of the entire error while variance error is low, we can assert that RMCLP is a stable but weak classifier, and boosting method can be used to improve its performance. On the other hand, if variance error takes the heaviest part of the entire error while bias error is low, we can say that RMCLP is a strong but unstable classifier, and under this circumstance, bagging can be exploited as the ensemble method.

The rest of this paper is organized as follows: in the next section, we give a short introduction of RMCLP model. In the third section, we introduce the bias variance decomposition method. In the fourth section, we introduce the ensemble methods and describe the bagging and adaboosting algorithms. In the fifth section, we carry out bias variance decomposition of RMCLP on synthetic datasets. In the last section, we finish our paper with several conclusions.

2 Regularized Multiple Criteria Linear Programming (RMCLP)

In this section, we will introduce the two groups RMCLP model. Since RMCLP model derives from the original MCLP model, we will introduce MCLP model first.

2.1 Multiple Criteria Linear Programming (MCLP) Model

MCLP model [11] is originally introduced for linearly separating two-group data sets. Assume a two-group data set A which has n instances $A=\{A_1, A_2, ..., A_n\}$, we define a boundary vector b to distinguish the first group G_1 and the second group G_2 by following the rules that, if an example $A_i \in G_1$, then $A_i x < b$; otherwise, $A_i x \geq b$. To formulate the criteria functions and complete constraints for data separation, some other variables need to be introduced. We define external measurement α_i to be the overlapping distance between boundary b and a training instance, say A_i. When A_i is wrongly classified, α_i will be equal to $|A_i x - b|$. We also define internal measurement β_i to be the distance of A_i from its adjusted boundary b^*. When A_i is correctly classified, distance β_i will equal to $|A_i x - b^*|$, where $b^* = b + \alpha_i$ or $b^* = b - \alpha_i$. To separate the two groups as far as possible, we design two objective functions which minimize the overlapping distances and maximize the distances between classes. Suppose $\|\alpha\|_p^p$ denotes for the relationship of all overlapping α_i while $\|\beta\|_q^q$ denotes for the aggregation of all distances β_i. The final correctly classified instances is depended on simultaneously minimize $\|\alpha\|_p^p$ and maximize $\|\beta\|_q^q$. By choosing $p=q=1$, we get the linear combination of these two objective functions as follows:

$$\text{(MCLP)} \qquad \textit{Minimize} \quad w_\alpha \sum_{i=1}^{n} \alpha_i - w_\beta \sum_{i=1}^{n} \beta_i \tag{1}$$

Subject to:

$$A_i x - \alpha_i + \beta_i - b = 0, \forall A_i \in G_1$$
$$A_i x + \alpha_i - \beta_i - b = 0, \forall A_i \in G_2$$

where A_i is given, x and b are unrestricted, α_i and $\beta_i \geq 0$.

2.2 RMCLP Model

A lot of empirical studies have shown that MCLP is a powerful tool for classification. However, there is no theoretical work on whether MCLP always can find an optimal solution under different kinds of training samples. To go over this difficulty, recently, Shi et.al [3] proposed a RMCLP model by adding two regularized items $x^T Hx/2$ and $\alpha^T Q\alpha/2$ on MCLP as follows:

$$\textbf{\textit{Minimize}} \ \frac{1}{2} x^T Hx + \frac{1}{2} \alpha^T Q\alpha + d^T \alpha - c^T \beta \tag{2}$$

Subjet to:

$$A_i x - \alpha_i + \beta_i = b, \forall A_i \in G_1;$$
$$A_i x + \alpha_i - \beta_i = b, \forall A_i \in G_2;$$
$$\alpha_i, \beta_i \geq 0.$$

where $H \in R^{r*r}$, $Q \in R^{n*n}$ are symmetric positive definite matrices. $d^T, c^T \in R^n$. The RMCLP model is a convex quadratic program. Theoretically studies [3] have shown that RMCLP can always find a global optimal solution. The algorithm of RMCLP is shown in Algorithm 1.

Algorithm 1. Building RMCLP model

Input: The data set $X = \{x_1, x_2, ..., x_n\}$, training percentage *p*
Output: RMCLP model *(w,b)*

Begin
1. Randomly select *p*/x/* instances as the training set ***TR***, the remained instances are combined as the testing set ***TS***;
2. Choose appropriate parameters of (H, D, d, c);
3. Apply the MCLP model (1) to compute the optimal solution $W^*=(w_1, w_2, \ldots, w_n)$ as the direction of the classification boundary;
4. Output $y=sgn(wx-b)$ as the model.

End

3 Bias Variance Decomposition

Notations. Consider a two groups classification problem with 0/1 loss function. Assume there is a set $D=\{D_1, D_2,\ldots,D_n\}$ of learning sets $D_j=\{(x_1,t_1),\ldots,(x_m,t_m)\}$, with $t_j \in C =\{-1,1\}$, and $x_j \in X$ has d attributes. The estimates of all the errors are performed on a test set T separated from the training set D. The learning algorithm is denoted by L. A classifier f_i built on D_i using L is denoted as $f_i = L(D_i)$. The predicted label of instance x_k by f_i is denoted as $y_k = f_i(x_k)$.

Measuring Bias and Variance. Bias-variance analysis provides a powerful tool to study learning algorithms and can be used to properly design ensemble methods well tuned to the properties of a specific base learner. Historically, the bias-variance analysis was borrowed from regression problems where squared-loss is often used [4]. Recently, Domingo proposed a unified framework [6] of bias-variance decomposition of error which can be used for an arbitrary loss function. According to Domingo's decomposition, the expected loss of a learner $\mathcal{L}$ on a test instance x can be decomposed as a noise error $N(x)$, a bias error $B(x)$ and a variance error $V(x)$ as follow,

$$E[L(\mathcal{L}, x)] = c_1 N(x) + B(x) + c_2 V(x) \tag{3}$$

where c_1 and c_2 are coefficients decided by the loss function. A good classifier should have low bias, in which case the expected loss will approximately equal the variance. To calculate (4), we give the definitions of *optimal prediction* and *main prediction* first. In presence of noise, instance x_i and instance x_j ($i \neq j$) may share the same attribute values but having different labels. We define the *optimal prediction* y^* on instance x to be the most frequent observed label t as

$$y^* = \text{argmax}_{t \in C}\, p(t \mid x). \tag{4}$$

We then define the *main prediction* y^m to be the most frequent class label that each base learner gives to x as follow,

$$y^m = \arg\max(p(c_1 \mid x), p(c_2 \mid x)). \tag{5}$$

By defining y^* and y^m, we can calculate the noise error of instance x by counting the number of discrepancies between class label t and y^* in (7),

$$N(x) = \sum_{t \in C} \| t \neq y^* \|\, p(t \mid x), \tag{6}$$

where $\| z \| = 1$ if z is true, otherwise 0. Then the bias error can be calculated by

$$B(x) = \left| \frac{y_m - t}{2} \right| = \begin{cases} 1, & y^m \neq y^* \\ 0, & y^m = y^* \end{cases} \tag{7}$$

The variance error is the discrepancies between each base classifier's prediction and the main prediction, which can be denoted as

$$V(x) = \frac{1}{s} \sum_{i=1}^{s} \| f(x_i) - y^m \| \tag{8}$$

where s is the number of base classifiers. Then the average bias, variance, and noise errors over the entire set of the examples in the test set $\mathcal{T}$ can be denoted as :

$$E_x[N(x)] = \frac{1}{n} \sum_{i=1}^{n} N(x_i) = \frac{1}{n} \sum_{i=1}^{n} \sum_{t} \| t_{x_i} \neq y^* \| p(t \mid x_i) \tag{9}$$

$$E_x[B(x)] = \frac{1}{n} \sum_{i=1}^{n} B(x_i) = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y^m - t_i}{2} \right|, \tag{10}$$

and

$$E_x[V(x)] = \frac{1}{n} \sum_{i=1}^{n} V(x_i) = \frac{1}{ns} \sum_{j=1}^{s} \sum_{i=1}^{n} \| y^m \neq f(x_i) \|. \tag{11}$$

where n is the number of examples in test set $\mathcal{T}$. Finally, the average loss on all the examples is the algebraic sum of the average bias, variance and noise errors as follow,

$$E_x[L(t, y)] = E_x[N(x)] + E_x[B(x)] + E_x[V(x)]. \tag{12}$$

4 Ensemble Methods

Ensemble of classifiers is one of the main research topics in machine learning. Empirical studies have shown that ensemble are often much more accurate than the individual base learner that makes them up in classification and regression problems. Two typical methods of ensemble strategy is bagging and boosting. Bagging can improve the performance of a learner by reducing its variance error, thus unstable learner such as C4.5 can achieve better prediction accuracy. In contrast, boosting can be used to

enhance the performance of a weak learner which is slightly better than random guess. By using boosting, a weak classifier can be lifted into a strong classifier. As we discussed above, bias variance decomposition offers a rationale to develop ensemble methods for RMCLP. If RMCLP is an unstable classifier with its error mainly coming from its variance, bagging can be used to transform RMCLP to be a stable classifier. On the other hand, if RMCLP is a weak classifier with its error mainly coming from its bias, boosting method can be used to enhance it into a strong classifier. Algorithm 2 shows the algorithm of Bagging and Algorithm 3 exhibits the Adaboosting algorithm (a refined version of boosting) [10].

Algorithm 2. Bagging Algorithm

Input: an unstable learner *L*, number of bags *b*,
training set $Tr = \{(x_1,y_1), \ldots, (x_n, y_n)\}$
Output: a stable leaner $f(x)$

Begin
1. for i = 1…b
Randomly select *n* samples from *X* with replacement to form a bag B_i;
Build a classifier using *L* on B_i to get classifier $C_i=L(B_i)$
end for
2. Combine the *n* base classifiers C_i to form an ensemble classifier:

$$f(x) = \sum_{i=1}^{n} C_i(x)$$

3. Output $f(x)$.
End

Algorithm 3. AdaBoosting Algorithm

Input: a weak Leaner *L*, maximal iteration number *T*,
training set $Tr = \{(x_1,y_1), \ldots, (x_n, y_n)\}$
Output: a strong leaner $f(x)$

Begin
1. Initialize distribution of *Tr* by $D_1(i)=1/N$;
2. for t = 1…T
2.1 Train *L* using D_t and get $f_t = L(D_t)$
2.2 Calculate w_t
2.3 update distribution by $D_{t+1}(i) = \dfrac{D_t(i)\, e^{-y_i w_t f_t(x_i)}}{Z_t}$, where Z_t is a normalizor.
end for
3. Output the ensemble learner (strong learner): $f(x) = \mathrm{sgn}(\sum_{i=1}^{T} w_i f_i(x))$
End

5 Bias Variance Decomposition to Develop Ensemble Method for RMCLP

We generate four synthetic datasets with levels of noise 0%, 5%, 10%, 15% respectively. Each dataset is a 2-dimensional 2-class problem with 600 instances, 300 for each class. 80% instances will be used for training and the remained 20% will be used as testing. All of the generated instances comply with the Gaussian distribution $x \sim N(\mu, \Sigma)$, where μ is the mean vector and Σ is the covariate matrix. The classification boundary is defined as a linear boundary $\| x \| = 1$. Instances that satisfy $\| x \| < 1$ will be assigned the class label "-1" while agree with $\| x \| \geq 1$ will be assigned the class label "+1". To simulate noisy instances, we randomly pick up 0.0%, 0.25%, 5%, 7.5% instances from each class and then assign them the opposite class label. In the following experiments, we will do bias-variance decomposition on these four datasets to investigate whether RMCLP is a stable and strong classifier.

Table 1 reports the 10-folder cross validation of bias variance results on the synthetic datasets. The first column lists three measurements, average error, bias error and variance error respectively. The second column shows the different parameters of RMCLP. The 3th to 6th column lists the experiment results under different training samples. In this experiments, we will do the following observations: (1) by using different parameters of *(H,D,d,c)*, we examine whether RMCLP varies significantly different on different parameters; (2) by comparing the bias and variance errors, we get a conclusion where the error mainly comes from, and thus we can define RMLCP as a strong or weak classifier, a stable or unstable classifier. However, we don't report the noise error *N(x)* here because noises error is independent of classifiers but only related to the nature of dataset. Additionally, we give approximate values to bias error by subtract variance error from the average error. This approximation is reasonable because it won't change their orders when compared. All of the numeric results are listed in Table 1.

From the results of average errors, we can observe that the error rates have slight changes when the parameters keep changing. On the 0% noise dataset, only when H>D, the error goes up from 0.003 to 0.005. On the other three datasets, these similar tendencies have been observed. Thus we can say that RMCLP is slightly impacted by different parameters.

From the variance error, we can observe that compared to the average errors and bias errors, the variance errors take a low percentage of the whole error. For example, on 0% noise dataset, most of the variance errors stay 0, that is to say, no changes have been observed. On the other three datasets, the variance error are almost 10^{-1} lower than the bias error. Thus it is safe to say that RMCLP is a stable classifier, and bagging will help little on RMCLP model.

From the bias error, we can see that on the four datasets, bias errors take part of most of the average errors. For example, on the 10% noise dataset, when H=D, d=c, the average error is 0.091, while the bias errors is 0.084, which takes up 92.3% of the whole error. Thus it is also safe to say that RMCLP is not a strong classifier, and boosting can be used to enhance RMCLP into a strong classifier.

Table 1. Results of Bias variance analysis

Measures	parameters	0%	5%	10%	15%
Avg. Error	H=D, d=c	0.003	0.050	0.091	0.116
	H=D, d>c	0.003	0.050	0.091	0.117
	H=D, d<c	0.003	0.050	0.090	0.116
	H>D, d=c	0.005	0.047	0.087	0.106
	H<D, d=c	0.003	0.048	0.085	0.116
Bias	H=D, d=c	0.003	0.042	0.084	0.107
	H=D, d>c	0.003	0.041	0.084	0.108
	H=D, d<c	0.003	0.042	0.085	0.108
	H>D, d=c	0.004	0.046	0.081	0.094
	H<D, d=c	0.003	0.045	0.082	0.107
Variance	H=D, d=c	0.000	0.008	0.007	0.009
	H=D, d>c	0.000	0.009	0.007	0.009
	H=D, d<c	0.000	0.008	0.005	0.008
	H>D, d=c	0.001	0.001	0.006	0.012
	H<D, d=c	0.000	0.003	0.003	0.009

6 Conclusions

In this paper, we applied bias-variance decomposition to RMCLP learning algorithms for the purposes of gaining a thorough understanding of RMCLP's stability and weakness for classification. Our theoretical and empirical studies have concluded that : (1) The parameter setting *(H,D,d,c)* has very limited impact on the performance of the RMCLP models; (2) The variance of the RMCLP models accounts for only a small part of the total testing error, which indicates that RMCLP is a stable classifier; (3) The testing error of RMCLP mostly comes from its bias, which suggests that RMCLP doesn't belong to the strong learner category; (4) Bagging has a limited effect on RMCLP as it is a stable learner; (5) By using boosting method (especially adaboosting), we can transform RMCLP into a strong classifier. In the future, we will use this bias variance decomposition to test RMCLP model on UCI benchmark datasets to further validate our conclusions.

References

1. Bennett, K.P., Parrado-Hernandez, E.: The Interplay of Optimization and Machine Learning Research. Journal of Machine Learning Research 7, 1265–1281 (2006)
2. Shi, Y., Wise, M., Luo, M., Lin, Y.: Data mining in credit card portfolio management: a multiple criteria decision making approach. Multiple Criteria Decision Making in the New Millennium, 427–436 (2001)
3. Shi, Y., Tian, Y., Chen, X., Zhang, P.: A Regularized Multiple Criteria Linear Program for Classification. In: ICDM Workshops 2007, pp. 253–258 (2007)
4. Kong, E.B., Dietterich, T.G.: Error-correcting output coding corrects bias and variance. In: Proc. of the 12th ICML Conference (1996)
5. Domingos, P.: A Unified Bias-Variance Decomposition and its Applications. In: Proc. of thseventeeth Interntional Conference on Machine Learning, Stanford, CA, pp. 231–238. Morgan Kaufmann, San Francisco (2000)

6. Domingos, P.: A Unifed Bias-Variance Decomposition for Zero-One and Squared Loss. In: Proc. of the Seventeenth National Conference on Artificial Intelligence, Austin, TX. AAAI Press, Menlo Park (2000)
7. Valentini, G., Dietterich, T.G.: Bias-Variance Analysis of Support Vector Machines for the Development of SVM-Based Ensemble Methods. Journal of Machine Learning Research 5, 725–775 (2004)
8. Breiman, L.: Bagging predictors. Journal of Machine Learning 24, 123–140 (1996)
9. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences 55(1), 119–139 (1997)
10. Polikar, R.: Ensemble Based Systems in Decision Making. IEEE Circuits and Systems Magazine 6(3), 21–45 (2006)
11. Peng, Y., Kou, G., Chen, Z., Shi, Y.: Cross-Validation and Ensemble Analyses on Multiple-Criteria Linear Programming Classification for Credit Cardholder Behavior. In: International Conference on Computational Science, pp. 931–939 (2004)

Appendix

In this appendix we give the deduction of the original bias variance decomposition in regression problems with squared-lose function.

Assume the genuine function we want to approximate is $f = f(x)$. There is a training sample $D = \{(x_i, t_i)\}_{i=1}^{N}$, where $x_i \in X$, for instance $X = \mathbb{R}^d$, $d \in N$ and $t_i \in T$ with $T = \mathbb{R}$. Besides, D contains a noise level ε with expectation of 0, that is to say, $t = f + \varepsilon$ and $E(\varepsilon) = 0$. A learning function $g=g(w,x)$ is used to approximate the genuine function f, where w is the parameters of g and x is the training sample. To a specific x_i, g gives a corresponding value $y_i=g(w,x_i)$. Thus, the mean squared loss of g on the whole training dataset D is

$$MSE = \frac{1}{|D|}\int_{x \in D} L(g,x)\,dx = \frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2 .$$

To investigate whether g is a good regression model on D, we calculate the expectation of the mean squared loss by

$$E\{MSE\} = E\{\frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2\} = \frac{1}{N}E\{\sum_{i=1}^{N}(t_i - y_i)^2\} . \tag{13}$$

To get $E\{MSE\}$, we calculate $E\left[(t_i - y_i)^2\right]$ as follows:

$$\begin{aligned} E\left[(t_i - y_i)^2\right] &= E\left[(t_i - f_i + f_i - y_i)^2\right] \\ &= E\left[(t_i - f_i)^2 + (f_i - y_i)^2 + 2(t_i - f_i)(f_i - y_i)\right] \\ &= E(t_i - f_i)^2 + E(f_i - y_i)^2 + 2E\left[(t_i - f_i)(f_i - y_i)\right] \\ &= E(\varepsilon_i)^2 + E(f_i - y_i)^2 + 2\left[\left(E(t_i f_i) - E(t_i y_i) - E(f_i^2) + E(f_i y_i)\right)\right] \\ &= E(\varepsilon_i^2) + E(f_i - y_i)^2 + 2\left[\left(E(t_i f_i) - E(t_i y_i) - E(f_i^2) + E(f_i y_i)\right)\right] \end{aligned} \tag{14}$$

As we discussed above, $t = f + \varepsilon$ and $E(\varepsilon) = 0$, thus we have the following two equations:

$$E(t_i f_i) = E(f_i^2 + \varepsilon_i f_i) = E(f_i^2), \tag{15}$$

$$E(t_i y_i) = E(f_i y_i + \varepsilon_i y_i) = E(f_i y_i). \tag{16}$$

Meanwhile

$$\begin{aligned} E(f_i - y_i)^2 &= E(f_i - E(y_i) + E(y_i) - y_i)^2 \\ &= E\left(f_i - E(y_i)\right)^2 + E\left(E(y_i) - y_i\right)^2 + 2E\left((f_i - E(y_i))(E(y_i) - y_i)\right) \\ &= E\left(f_i - E(y_i)\right)^2 + E\left(E(y_i) - y_i\right)^2 + 2\left(E(f_i)E(y_i) - E(f_i y_i) - E^2(y_i) + E^2(y_i)\right) \\ &= E\left(f_i - E(f_i)\right)^2 + E\left(E(f_i) - y_i\right)^2 \end{aligned} \tag{17}$$

Consequently, by combining (14), (15), (16) and (17), we can get the result of (13) as follows:

$$\begin{aligned} E\{MSE\} &\propto E\left[(t_i - y_i)^2\right] \\ &= E(\varepsilon_i^2) + E(f_i - y_i)^2 \\ &= \underbrace{E(\varepsilon_i^2)}_{noise} + \underbrace{E\left(f_i - E(y_i)\right)^2}_{Bias^2} + \underbrace{E\left(E(y_i) - y_i\right)^2}_{Variance} \end{aligned}$$

We can see that under this decomposition, the average error of a learner *g* is composed of its noise, bias and variance errors. Noise error is the inherent characteristics of the training sample that we can do nothing to improve. Bias error is the error between the target function *f* and the average output of learning algorithm *g*. Variance error is the error between the individual output of *g* and the average output of *g*.

Knowledge-Rich Data Mining in Financial Risk Detection

Yi Peng[1,2], Gang Kou[1,2,*], and Yong Shi[2,3]

[1] School of Management and Economics, University of Electronic Science and Technology of China, Chengdu, P.R. China, 610054
[2] CAS Research Center on Fictitious Economy and Data Sciences, Beijing 100080, China
[3] College of Information Science & Technology, University of Nebraska at Omaha, Omaha, NE 68182, USA
kougang@uestc.edu.cn

Abstract. Financial risks refer to risks associated with financing, such as credit risk, business risk, debt risk and insurance risk, and these risks may put firms in distress. Early detection of financial risks can help credit grantors to reduce risk and losses, establish appropriate policies for different credit products and increase revenue. As the size of financial databases increases, large-scale data mining techniques that can process and analyze massive amounts of electronic data in a timely manner become a key component of many financial risk detection strategies and continue to be a subject of active research. However, the knowledge gap between the results data mining methods can provide and actions can be taken based on them remains large in financial risk detection. The goal of this research is to bring the concept of chance discovery into financial risk detection to build the knowledge-rich data mining process and therefore increase the usefulness of data mining results in financial risk detection. Using six financial risk related datasets, this research illustrates that the combination of data mining techniques and chance discovery can provide knowledge-rich data mining results to decision makers; promote the awareness of previously unnoticed chances; and increase the actionability of data mining results.

Keywords: knowledge-rich data mining, chance discovery, financial risk detection, classification, risk analysis.

1 Introduction

Financial risks refer to risks associated with financing, such as credit risk, business risk, debt risk and insurance risk, and these risks may put firms in distress. Take health insurance risk as an example. According to the National Health Care Anti-Fraud Association's (NHCAA) estimation, at least 3% of the United States' annual health care expenditure, which in calendar-year 2003 alone amounted to $1.7 trillion, is lost to outright fraud or erroneous payment [20]. Another example is credit card debt risk. The total credit card debt at the end of the first quarter of 2002 in the U.S. is about $660 billion [21] and the total credit card holders declared bankruptcy in 2003 are more than 1.6 million [14].

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 534–542, 2009.

As the size of financial databases increases, large-scale data mining techniques that can process and analyze massive amounts of electronic data in a timely manner become a key component of many financial risk detection strategies and continue to be a subject of active research [4, 11, 16, 17]. While most data mining research focus on developing algorithms or applying data mining methods to detect financial risk, how to better utilize data mining results to help decision makers identify financial risks and integrate data mining results into decision-making routines have not been widely studied in the data mining community. The lack of interaction between industry practitioners and academic researchers makes it hard to discover financial risks or opportunities in data mining projects and hence weaken the value that data mining methods may bring to financial risk detection. The knowledge gap between the results data mining methods can provide and taking actions based on them [26] remains large in financial risk detection.

The goal of this paper is to bring the concept of chance discovery into financial risk detection to build the knowledge-rich data mining process and therefore increase the usefulness and actionability of data mining results in financial risk detection. The rest of this paper is organized as follows: Section 2 gives an overview of the datasets employed for this study; discusses the data mining techniques used in financial risk detection; and proposes a knowledge-rich data mining process for financial risk detection. Section 3 presents an empirical study that examines the proposed process using six financial risk datasets and Section 4 summarizes the paper.

2 Financial Risk Detection Problem

There is no one universally accepted definition for financial risk. Any risk that relates to financing can be considered as financial risk. According to its sources, financial risk can be broadly categorized as investment risk, credit risk, and business risk [24]. Investment risk is the probability that an investment may produce an undesirable outcome. Credit risk is the probability that debtors will not pay their debts. Business risk denotes the possibility that income is less than expected and/or expenditure is larger than expected. The datasets used in this work are examples of credit and business risk: consumer credit card application and existing credit cardholders' behavior belong to credit related risk; corporation bankruptcy and disability income insurance datasets are business related risks. Financial risk detection is the practice of identifying or predicting these financial risks in an attempt to control risk and minimize losses. The following subsections describe the data sources and examine the major data mining techniques that have been applied in this work.

2.1 Financial Risk Datasets

The datasets used in this study come from five countries and represent four aspects of financial risk: credit approval (credit card application and loan approval), credit behavior, bankruptcy risk, and fraud risk.

German credit card application dataset [22]
The German credit card application dataset comes from UCI Machine Learning databases. It contains 1000 instances with 24 predictor variables and 1 class variable.

The 24 variables describe the status of existing checking account, credit history, education level, employment status, personal status, age, and so on. The class variable indicates whether an application is Accepted or Declined. 70% of the instances are accepted applications and 30% are declined instances.

Australian credit card application dataset [19]
This dataset was provided by a large bank and concerns consumer credit card applications. It has 690 instances with 15 predicator variables plus 1 class variable. The class variable indicates whether an application is Accepted or Declined. 55.5% of the instances are accepted applications and 44.5% are declined instances.

Credit cardholders' behavior dataset [1, 11, 16]
The dataset was from a major US bank and contains 6000 credit card data with 64 predictor variables plus 1 class variable. Each instance has a class label indicating its credit status: either Good or Bad. 84% of the data are Good accounts and 16% are Bad accounts. Good indicates good status accounts and Bad indicates accounts with late payments, delinquency, or bankruptcy. The predictor variables describe account balance, purchase, payment, cash advance, interest charges, date of last payment, times of cash advance, and account open date.

Japanese bankruptcy dataset [12]
This set collects 37 bankrupt Japanese firms and 111 non-bankrupt Japanese firms from various sources during the post-deregulation period of 1989 to 1999. Final sample firms are ones traded in the First Section of Tokyo Stock Exchange, and their financial data are available from 2000 PACAP database for Japan compiled by the Pacific-Basin Capital Market (PACAP) Research Center at the University of Rhode Island. Each case has 13 predictor variables and 1 class variable (Bankrupt or Non-bankrupt). The predictor variables describe financial state and performance of firms.

Korean bankruptcy dataset [13]
This dataset collects bankrupt firms in Korea from 1997 to 2003 from public sources. It consists of 65 bankrupt and 130 non-bankrupt firms whose data are available and publicly trading firms in the Korean Stock Exchange. Each case has 13 predictor variables with one class variable (Bankrupt or Non-bankrupt).

Disability income insurance dataset [17]
Disability insurance provides financial support when a disability leaves the policyholder unable to work. The data was provided by an anonymous U.S. corporation. Each instance concerns a disability income insurance claim. The set has 18,875 instances with 103 variables. A binary class attribute indicates whether an instance is a Normal claim or Abnormal claim. There are 353 abnormal claims and 18,522 normal claims. The abnormal instances represent fraudulent or erroneous claims and were manually collected and verified.

2.2 Data Mining Techniques

Current data mining research in financial risk detection concentrates on fraud detection and credit risk analysis. Major data mining functions used in these areas include classification, prediction, cluster analysis and outlier analysis. The selection of data mining functions depends on data mining tasks. Since the datasets employed in this

study are examples of classification applications, the data mining function used in this paper is classification. Eight classification methods: Bayesian network [23], naïve Bayes [8], Support Vector Machine (SVM) [18], linear logistic regression [3], K-nearest-neighbor [7], C4.5 [19], Repeated Incremental Pruning to Produce Error Reduction (RIPPER) rule induction [6] and radial basis function (RBF) network [2], are used in our empirical study and all of them are implemented in WEKA [25].

Bayesian network and naïve Bayes classifier both model probabilistic relationships between predictor variables and the class variable. While naïve Bayes classifier estimates the class-conditional probability based on Bayes theorem and can only represent simple distributions, Bayesian network is a probabilistic graphic model and can represent conditional independencies between variables. SVM classifier uses a nonlinear mapping to transform the training data into a higher dimension and search for the linear optimal separating hyperplane, which is then used to separate data from different classes [10]. Linear logistic regression models the probability of occurrence of an event as a linear function of a set of predictor variables. K-nearest-neighbor classifies a given data instance based on learning by analogy, that is, assigns it to the closest training examples in the feature space. C4.5 is a decision tree algorithm that constructs decision trees in a top-down recursive divide-and-conquer manner. RIPPER is a sequential covering algorithm that extracts classification rules directly from the training data without generating a decision tree first [10]. RBF network is an artificial neural network that uses radial basis functions as activation functions. In addition to the eight classification techniques, we also employ ensemble method, which aggregates the predictions of the above mentioned eight classifiers.

2.3 Knowledge-Rich Data Mining in Financial Risk Detection

Even though data mining has become a crucial tool in financial risk detection, most data mining research focus on developing learning algorithms or improving existing algorithms that can identify suspicious patterns or predict future behaviors efficiently from financial databases and have not paid enough attention to the involvement of end users and the actionability of the final data mining results. This is mainly due to two reasons: (1) the difficulty in accessing real-life financial risk data; (2) limited access to domain experts and background information. The lack of interaction between industry practitioners and academic researchers makes it hard to discover financial risks or opportunities in data mining projects and hence weaken the value that data mining methods may bring to financial risk detection.

In an attempt to improve the usefulness of data mining results and increase the probability of identifying unusual chances in financial risk analysis, this paper proposes a chance discovery and data mining process for financial risk detection (Figure 1). Chance discovery (CD) is defined as "the awareness of a chance and the explanation of its significance" [15]. Ohsawa and Fukuda [15] suggested three keys to chance discovery: communicating the significance of an event; enhancing user's awareness of an event's utility using mental imagery; revealing the causalities of rare events using data mining methods. Users, communication and data mining are the main parts of chance discovery. Figure 1 combines the KDD (Knowledge Discovery in Database) process model [9], the chance discovery process [15] and the CRISP-DM process model [5]. It emphasizes three keys to chance discovery and knowledge-rich data

mining: users, communication and data mining techniques. Users refer to domain experts and decision makers. Domain experts are knowledgeable of the field information, data collection procedures and meaning of variables. With the assistance of data miners, domain experts can gain insights of financial risk data from different aspects and potentially observe new chances. To turn the identified knowledge into financial or strategic advantages, decision makers, who understand the operational and strategic goals of a company, are required to provide feedbacks on the importance of the potential new chances and determine what actions should be taken. Moving back and forth between steps is always required. The cyclical nature of chance discovery is illustrated by the outer circle of the chance discovery process in Figure 1.

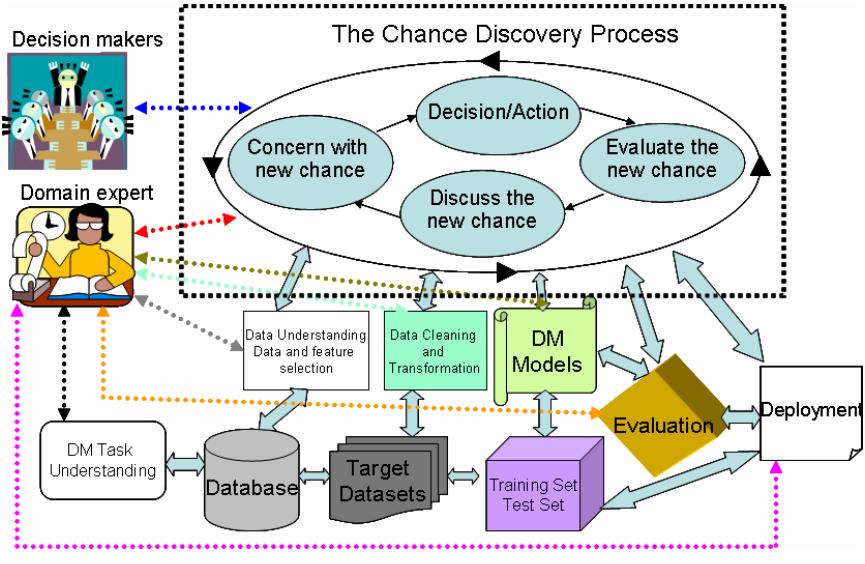


Fig. 1. The process of chance discovery and data mining for financial risk detection

3 Empirical Study

In this section we apply the proposed process in Section 2 to the six financial risk datasets. The datasets were collected from five countries and represent four aspects of financial risk: credit approval, credit behavior, bankruptcy risk, and insurance fraud. The experiment was carried out according to the following process:

Chance discovery process for financial risk detection

Input: a financial risk related dataset

Output: Decision function; Results of performance metrics; new chance(s)

Step 1. Understand business requirements, dataset structure and data mining task

Step 2. Prepare target datasets: select and transform relevant features; data cleaning; data integration. Communicate any findings during data preparation to domain experts.

Step 3. Train and test multiple data mining models in randomly sampled partitions (e.g. k-fold cross-validation) using WEKA 3.5.7 [25].

Step 4. Evaluate data mining models using a set of performance metrics. The best model/s is the decision function.

Step 5. Discuss the data mining results with domain experts. Explore potential chance(s) from data mining results. If identify new chance(s), communicate the chance(s) with decision makers and determine the appropriate actions. Go back to Step 1 if new business questions are raised during the process.

END

Because different performance metrics are appropriate in different settings, this paper utilizes six performance metrics: True Positive rate, True Negative rate, False Positive rate, False Negative rate, Overall Accuracy and Area Under ROC. TP (True Positive) is the number of correctly classified Abnormal (Bankrupt, Bad, or Declined) instances. FP (False Positive) is the number of Normal (Non-bankrupt, Good, or Accepted) instances that is misclassified as Abnormal class. TN (True Negative) is the number of correctly classified Normal instances. FN (False Negative) is the number of Abnormal instances that is misclassified as Normal class. Accuracy is one the most widely used classification performance metrics. It is the ratio of correctly predicted instances to the entire instances or instances in a particular class.

$$\text{Overall Accuracy} = \frac{TN + TP}{TP + FP + FN + TN}$$

$$\text{True Positive rate} = \frac{TP}{TP + FN}, \text{True Negative rate} = \frac{TN}{TN + FP}$$

$$\text{False Positive rate} = \frac{FP}{FP + TN}, \text{False Negative rate} = \frac{FN}{FN + TP}$$

ROC curves stands for Receiver Operating Characteristic which detects and characterizes the tradeoff between TP rate and FP rate. The area under the curve represents the probability of the ranking by the classifier of a randomly sampled positive instance over a randomly sampled negative one. It indicates a classifier's performance; the larger the area, the better the classifier.

The classification results of eight data mining methods plus ensemble method for the six datasets are summarized in Table 1. In the dataset column, Australian indicates Australian credit card application data; Chase indicates_credit cardholders' behavior data; DI indicates the disability income insurance data; German indicates German credit card application data; Japan indicates Japanese bankruptcy data and Korea indicates Korean bankruptcy data. The nine classification methods described in Section 2 were applied to each dataset using 10-fold cross-validation.

Table 1. Classification results

Dataset	Algorithm	Overall Accuracy	Area Under ROC	True Positive rate	True Negative rate	False Positive rate	False Negative rate
Australian	Bayesian Network	0.8522	0.9143	0.7980	0.8956	0.1044	0.2020
Australian	Naive Bayes	0.7725	0.8978	0.5863	0.9217	0.0783	0.4137
Australian	SVM	0.8551	0.8622	0.9251	0.7990	0.2010	0.0749
Australian	Linear Logistic	0.8623	0.9312	0.8664	0.8590	0.1410	0.1336
Australian	K Nearest Neighbor	0.7942	0.7922	0.7752	0.8094	0.1906	0.2248
Australian	C4.5	0.8348	0.8346	0.7948	0.8668	0.1332	0.2052
Australian	RBFNetwork	0.8304	0.8995	0.7524	0.8930	0.1070	0.2476
Australian	RIPPER Rule Induction	0.8522	0.8714	0.8534	0.8512	0.1488	0.1466
Australian	Ensemble	0.8551	0.9289	0.8274	0.8773	0.1227	0.1726
Chase	Bayesian Network	0.7055	0.8424	0.8656	0.6750	0.3250	0.1344
Chase	Naive Bayes	0.6933	0.8395	0.8740	0.6589	0.3411	0.1260
Chase	SVM	0.8372	0.5632	0.1604	0.9661	0.0339	0.8396
Chase	Linear Logistic	0.8532	0.8539	0.3031	0.9579	0.0421	0.6969
Chase	K Nearest Neighbor	0.8028	0.6327	0.3802	0.8833	0.1167	0.6198
Chase	C4.5	0.8170	0.6245	0.3542	0.9052	0.0948	0.6458
Chase	RBFNetwork	0.8400	0.8256	0.0000	1.0000	0.0000	1.0000
Chase	RIPPER Rule Induction	0.8443	0.6380	0.3333	0.9417	0.0583	0.6667
Chase	Ensemble	0.8382	0.8432	0.3990	0.9218	0.0782	0.6010
DI	Bayesian Network	0.8261	0.8361	0.6686	0.8291	0.1709	0.3314
DI	Naive Bayes	0.9695	0.9707	0.9943	0.9691	0.0309	0.0057
DI	SVM	0.9813	0.5000	0.0000	1.0000	0.0000	1.0000
DI	Linear Logistic	0.9809	0.7546	0.0000	0.9996	0.0004	1.0000
DI	K Nearest Neighbor	0.9723	0.5961	0.2040	0.9870	0.0130	0.7960
DI	C4.5	0.9786	0.6656	0.1898	0.9937	0.0063	0.8102
DI	RBFNetwork	0.9813	0.7097	0.0000	1.0000	0.0000	1.0000
DI	RIPPER Rule Induction	0.9806	0.5774	0.1586	0.9962	0.0038	0.8414
DI	Ensemble	0.9817	0.8443	0.0765	0.9989	0.0011	0.9235
German	Bayesian Network	0.7250	0.7410	0.3600	0.8814	0.1186	0.6400
German	Naive Bayes	0.7550	0.7888	0.5067	0.8614	0.1386	0.4933
German	SVM	0.7740	0.6938	0.4933	0.8943	0.1057	0.5067
German	Linear Logistic	0.7710	0.7919	0.4933	0.8900	0.1100	0.5067
German	K Nearest Neighbor	0.6690	0.6064	0.4500	0.7629	0.2371	0.5500
German	C4.5	0.7190	0.6607	0.4400	0.8386	0.1614	0.5600
German	RBFNetwork	0.7400	0.7520	0.4633	0.8586	0.1414	0.5367
German	RIPPER Rule Induction	0.7340	0.6557	0.4500	0.8557	0.1443	0.5500
German	Ensemble	0.7620	0.7980	0.4533	0.8943	0.1057	0.5467
Japan	Bayesian Network	0.7568	0.7292	0.5135	0.8378	0.1622	0.4865
Japan	Naive Bayes	0.7432	0.7197	0.4595	0.8378	0.1622	0.5405
Japan	SVM	0.7500	0.5000	0.0000	1.0000	0.0000	1.0000
Japan	Linear Logistic	0.7770	0.7290	0.4595	0.8829	0.1171	0.5405
Japan	K Nearest Neighbor	0.7770	0.6595	0.4324	0.8919	0.1081	0.5676
Japan	C4.5	0.7162	0.5270	0.3784	0.8288	0.1712	0.6216
Japan	RBFNetwork	0.7162	0.6533	0.2162	0.8829	0.1171	0.7838
Japan	RIPPER Rule Induction	0.7365	0.6193	0.4324	0.8378	0.1622	0.5676
Japan	Ensemble	0.7905	0.7424	0.3243	0.9459	0.0541	0.6757
Korea	Bayesian Network	0.8667	0.8773	0.7846	0.9077	0.0923	0.2154
Korea	Naive Bayes	0.7744	0.8168	0.5538	0.8846	0.1154	0.4462
Korea	SVM	0.8718	0.8682	0.8615	0.8769	0.1231	0.1385
Korea	Linear Logistic	0.8462	0.8749	0.7692	0.8846	0.1154	0.2308
Korea	K Nearest Neighbor	0.8154	0.7993	0.7538	0.8462	0.1538	0.2462
Korea	C4.5	0.8359	0.7948	0.7077	0.9000	0.1000	0.2923
Korea	RBFNetwork	0.8256	0.8033	0.7231	0.8769	0.1231	0.2769
Korea	RIPPER Rule Induction	0.8667	0.8577	0.8308	0.8846	0.1154	0.1692
Korea	Ensemble	0.8564	0.9026	0.8154	0.8769	0.1231	0.1846

To summarize, the empirical study demonstrated that introducing chance discovery into the KDD process can promote the awareness of previously unnoticed chances and increase the usefulness of data mining results.

4 Conclusion Remarks

In financial risk detection, fail to recognize significant chances may cause credit grantors and companies serious financial losses. In order to capture significant chances in financial risk detection, this paper proposed a knowledge-rich data mining process model and tested the model using real-life financial risk related datasets. The empirical study indicated that the combination of data mining techniques and chance discovery can provide human inspectors more information than data mining alone; promotes the awareness of identified financial risks among decision makers; and increases the usefulness of data mining results in the decision-making process. Due to the space limitation, this paper only reported the application of the knowledge-rich data mining process model to fraud detection in insurance data. In the future, we will continue the study about the usability of the process in other areas of financial risk detection, such as bankruptcy prediction and credit cardholders' behavior analysis.

Acknowledgements

This work was supported by the Youth Fund of University of Electronic Science and Technology of China (UESTC) and the National Natural Science Foundation of China (NSFC) under the Grand No. 70621001, No. 70531040, and No. 70472074 and 973 Project #2004CB720103, Ministry of Science and Technology, China.

References

1. Peng, Y., Kou, G., Shi, Y., Chen, Z.: Improving Clustering Analysis for Credit Card Accounts Classification. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3516, pp. 548–553. Springer, Heidelberg (2005)
2. Bishop, C.M.: Neural networks for pattern recognition. Oxford University Press, Oxford (1995)
3. le Cessie, S., Houwelingen, J.C.: Ridge estimators in logistic regression. Applied Statistics 41(1), 191–201 (1992)
4. Chan, P.K., Fan, W., Prodromidis, A.L., Stolfo, S.J.: Distributed data mining in credit card fraud detection. IEEE Intelligent Systems 14(6), 67–74 (1999)
5. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R.: CRISP-DM 1.0: Step-by-Step data mining guide (2000), `http://www.crisp-dm.org`
6. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 115–123. Morgan Kaufmann, San Francisco (1995)
7. Dasarathy, B.V.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (1991)

8. Domingos, P., Pazzani, M.: On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. Machine Learning 29(203), 103–130 (1997)
9. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P.: The KDD Process for Extracting Useful Knowledge from Volumes of Data. Communications of the ACM 39(11), 27–34 (1996)
10. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2006)
11. Kou, G., Peng, Y., Shi, Y., Wise, M., Xu, W.: Discovering Credit Cardholders' Behavior by Multiple Criteria Linear Programming. Annals of Operations Research 135(1), 261–274 (2005)
12. Kwak, W., Shi, Y., Eldridge, S., Kou, G.: Bankruptcy Prediction for Japanese Firms: Using Multiple Criteria Linear Programming Data Mining Approach. International Journal of Business Intelligence and Data Mining 1(4), 401–416 (2006)
13. Kwak, W., Shi, Y., Kou, G.: Bankruptcy Prediction for Korean Firms after, Financial Crisis: Using Multiple Criteria Linear Programming Data Mining Approach (working paper) (1997)
14. New Generation Research, Inc., http://www.bankruptcydata.com/default.asp (as of April 27, 2008)
15. Ohsawa, Y., Fukuda, H.: Chance discovery by stimulated groups of people-application to understanding consumption of rare food. Journal of Contingencies and Crisis Management 10(3) (September 2002)
16. Peng, Y., Kou, G., Shi, Y., Chen, Z.: A Multi-Criteria Convex Quadratic Programming Model for Credit Data Analysis. Decision Support Systems 44(4), 1016–1030 (2008)
17. Peng, Y., Kou, G., Sabatka, A., Matza, J., Chen, Z., Khazanchi, D., Shi, Y.: Application of Classification Methods to Individual Disability Income Insurance Fraud Detection. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007, Part III. LNCS, vol. 4489, pp. 852–858. Springer, Heidelberg (2007)
18. Platt, J.C.: Fast training of support vector machines using sequential minimal optimization. In: Schotolkopf, B., Burges, C.J.C., Smola, A. (eds.) Advances in Kernel Methods-Support Vector Learning, pp. 185–208. MIT press, Cambridge (1998)
19. Quinlan, J.R.: C4.5: Programs for machine learning. Morgan Kaufmann, San Francisco (1993)
20. The National Health Care Anti-Fraud Association, http://www.nhcaa.org/eweb/DynamicPage.aspx?webcode=anti_fraud_resource_centr&wpscode=TheProblemOfHCFraud (as of April 27, 2008)
21. The U.S. Payment Card Information Network, http://www.cardweb.com/cardlearn/stat.html (as of April 27, 2008)
22. UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA, http://www.ics.uci.edu/~mlearn/MLRepository.html
23. Weiss, S.M., Kulikowski, C.A.: Computer Systems that Learn: Classification and Predication Methods from Statistics. In: Neural Nets Machine Learning and Expert Systems. Morgan Kaufmann, San Francisco (1991)
24. Wikipedia.org, http://en.wikipedia.org/wiki/Financial_risk (as of April 28, 2008)
25. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)
26. Domingos, P.: Toward knowledge-rich data mining. Data Mining and Knowledge Discovery 15, 21–28 (2007)

Smoothing Newton Method for L_1 Soft Margin Data Classification Problem

Weibing Chen[1], Hongxia Yin[2], and Yingjie Tian[1]

[1] Research Center on Fictitious Economy & Data Science,
Chinese Academy of Sciences, Beijing 100190, China
[2] Department of Mathematics and Statistics, Minnesota State University Mankato,
273 Wissink Hall, Mankato, MN 56001, USA
chenweibing06@mails.gucas.ac.cn,
hongxia.yin@mnsu.edu,
tianyingjie1213@163.com

Abstract. A smoothing Newton method is given for solving the dual of the l_1 soft margin data classification problem. A new merit function was given to handle the high-dimension variables caused by data mining problems. Preliminary numerical tests show that the algorithm is very promising.

1 Introduction

In the multiple kernel learning two-class nonlinear data classification problem, we suppose that n data $\{(x_i, y_i)\}$ are given, where $x_i \in \mathcal{X}$ for some input space $\mathcal{X} \subset \mathbb{R}^p$, and $y_i \in \{-1, 1\}$ indicating the class to which the point x_i belongs and are called lables. We assume that $\mathcal{F}$ is an embedding space (called feature space), φ is a map from $\mathcal{X}$ to $\mathcal{F}$. A kernel is a function k such that $k(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$ for any $x_i, x_j \in \mathcal{X}$, where $\langle \cdot, \cdot \rangle$ is the inner production. A kernel matrix is a square matrix $K \in \mathbb{R}^{n \times n}$ such that $K_{ij} = k(x_i, x_j)$ for $x_1, \cdots, x_n \in \mathcal{X}$. It is known that every kernel matrix is symmetric positive semidefinite. Kernel based method for two-class nonlinear classification is to find an affine function in the feature space, $f(x) = \langle w, \varphi(x) \rangle + b$ for some weight vector $w \in \mathcal{F}$ and $b \in \mathbb{R}$ to maximize the margin (or distance) between the parallel hyperplanes $\langle w, \varphi(x) \rangle + b = 1$ and $\langle w, \varphi(x) \rangle + b = -1$ in the high-dimensional feature space that are as far apart as possible while still separating the data. The hard margin problem is to find (w^*, b^*) that solves

$$\begin{aligned} &\min_{w,b} \ \langle w, w \rangle \\ &s.t. \ y_i(\langle w, \Phi(x_i) \rangle + b) \geq 1, \ i = 1, \cdots, n, \end{aligned} \tag{1}$$

which realizes the maximal margin classifier with geometric margin $\gamma = \frac{2}{\|w^*\|}$, assuming it exists. γ is actually the distance between the convex hulls of the two classes of data.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 543–551, 2009.

Problem (1) was widely investigated by many authors, see [1], [9], [10], [11], [13], and [14] for example. However, since the solution of (1) exists only when the labeled sample is linearly separable in the feature space ($f(x) = \langle w, \Phi(x)\rangle + b$), the following soft margin problem was defined by introducing slack variable $\xi = (\xi_1, \xi_2, \cdots, \xi_n)^T$ with $\xi_i \geq 0$ for all $i = 1, \cdots, n$ in order to relax the constraints in (1) to

$$y_i(\langle w, \Phi(x)\rangle + b) \geq 1 - \xi_i, \quad i = 1, \cdots, n.$$

There are variety of ways to define the soft margin problem [4], [9], [10], [11]. In 2004, Ferris and Munson [5] produced a semismooth support vector machine (SVM) where the soft margin was defined by the least-square of the error (it is l_2 norm). In the paper, We consider the l_1-norm soft margin problem

$$\begin{aligned} \min_{w,b,\xi} \quad & \langle w, w\rangle + C\sum_{i=1}^{n} \xi_i \\ s.t.\ & y_i(\langle w, \Phi(x)\rangle + b) \geq 1 - \xi_i, \quad i = 1, \cdots, n, \\ & \xi_i \geq 0, \quad i = 1, \cdots, n, \end{aligned} \tag{2}$$

where the parameter $C > 0$ determines the trade off between a large margin $\gamma = 2/\|w\|$ and a small error penalty. The dual problem of (2) is

$$\begin{aligned} \max\ & 2\alpha^T e - \alpha^T G(K)\alpha \\ \text{s.t.}\ & y^T\alpha = 0, \\ & 0 \leq \alpha \leq \mathbf{C}, \end{aligned} \tag{3}$$

Here $\mathbf{C}$ is a constant vector with all components to be the same positive number C, $\alpha \in \mathbb{R}^n$ is the dual parameter with its components $\alpha_i (i = 1, 2, \cdots, n)$ taking values in the interval $[0, C]$. For a fixed kernel K, (3) gives an upper bound on misclassification probability (see [1] Chapter 4 for details), solving problem (3) for a single kernel matrix is therefore a way to minimize the upper bound on error probability.

Smoothing Newton method was verified to be an efficient way to some nonsmooth equation system with global convergence and local superlinear convergence. Especially, it is successfully used to solve variation inequality, complementarity problems and KKT system of optimization problems. See [6], [3], and [8]. In the paper, we introduce a smoothing Newton method for solving the dual of the l_1 soft margin SVM problem (3). Moreover, in order to overcome the difficulty caused by the high-dimensional variables in data mining, we introduce a new merit function based on the Huber function [7].

The paper is organized as follows. In Section 2, we produce the smoothing equations for the KKT system of problem (3) by projection technique, then we introduce a new merit function and give a smoothing Newton method for solving (3). Numerical tests for illustrating the efficiency of the algorithm are given in Section 3. Conclusion and further remarks are in Section 4.

2 Smoothing Newton Method

In this section, we introduce a smoothing Newton method for solving (3). For this we rewrite the problem as the following minimization problem.

$$\begin{aligned} \min\ & \tfrac{1}{2}\alpha^T G(K)\alpha - e^T\alpha \\ \text{s.t.}\ & y^T\alpha = 0, \\ & 0 \le \alpha \le \mathbf{C}. \end{aligned} \tag{4}$$

The Lagrange function for problem (4) is

$$L(\alpha,\lambda) = \frac{1}{2}\alpha^T G(K)\alpha - e^T\alpha - \lambda y^T\alpha \tag{5}$$

and its derivative $L'_\alpha(\alpha,\lambda) = G(K)\alpha - e - \lambda y, L'_\lambda(\alpha,\lambda) = -y^T\alpha$. We denote $u = (\alpha,\lambda) \in \mathbb{R}^n \times \mathbb{R}$ and

$$F(u) = F(\alpha,\lambda) = \begin{bmatrix} G(K)\alpha - \lambda y - e \\ y^T\alpha \end{bmatrix}. \tag{6}$$

Then α^* is a solution of (4) if and only if there exists λ^* such that $u^* = (\alpha^*,\lambda^*)$ is a solution of the following variational inequality problem

$$(u-u^*)^T F(u^*) \ge 0, \forall\ u = (\alpha,\lambda) \in \Omega = [0,C]^n \times \mathbb{R}. \tag{7}$$

It is well known from [6] that solving (7) is equivalent to finding a root of the Robinson's Normal equations:

$$F(\Pi_\Omega(u)) + u - \Pi_\Omega(u) = 0 \tag{8}$$

where $\Pi_\Omega(u)$ is the projection of u onto Ω. From the definition of Ω we have that

$$\begin{bmatrix} G(K)\Pi_{[0,C]^n}\alpha - \lambda y - e \\ y^T\Pi_{[0,C]^n}\alpha \end{bmatrix} + \begin{bmatrix} \alpha \\ \lambda \end{bmatrix} - \begin{bmatrix} \Pi_{[0,C]^n}\alpha \\ \lambda \end{bmatrix} = 0, \tag{9}$$

that is,

$$H(u) := H(\alpha,\lambda) := \begin{bmatrix} G(K)\Pi_{[0,C]^n}\alpha + \alpha - \Pi_{[0,C]^n}\alpha - \lambda y - e \\ y^T\Pi_{[0,C]^n}\alpha \end{bmatrix} = 0. \tag{10}$$

Recall that for any three numbers $c \in \mathbb{R} \cup \{-\infty\}$, $d \in \mathbb{R} \cup \{\infty\}$ with $c \le d$ and $v \in \mathbb{R}$, the median function

$$\mathrm{mid}(c,d,v) = \Pi_{[c,d]}(v) = \begin{cases} c & \text{if } v < c \\ v & \text{if } c \le v \le d \\ d & \text{if } d < v \end{cases}$$

and the Chen-Harker-Kanzow-Smale smoothing function for $\mathrm{mid}(c,d,v)$ is

$$\phi(t,c,d,v) = \frac{c + \sqrt{(c-v)^2 + 4t^2}}{2} + \frac{d - \sqrt{(d-v)^2 + 4t^2}}{2}, \tag{11}$$

where $(t, v) \in \mathbb{R}_{++} \times \mathbb{R}$. It can be seen that function $\phi(\cdot)$ is continuously differentiable at any $(t, v) \in \mathbb{R}_{++} \times \mathbb{R}$.

By defining $\phi : \mathbb{R}_{++} \times \mathbb{R}^n$ with its components

$$\phi_i(t, \alpha) = \phi(t, 0, C, \alpha_i) = \frac{\sqrt{(\alpha_i)^2 + 4t^2}}{2} + \frac{C - \sqrt{(C - \alpha_i)^2 + 4t^2}}{2}, \quad (12)$$

where $(t, \alpha_i) \in \mathbb{R}_{++} \times \mathbb{R}, i = 1, 2, \cdots, n$, we have the smoothing equations associated to equations (10) as follows

$$\begin{array}{l} \sum_{j=1}^{n} G(K)_{ij}\phi_j(t, \alpha) + \alpha_i - \phi_i(t, \alpha) - \lambda y_i - 1 = 0, \quad i = 1, \cdots, n, \\ \sum_{j=1}^{n} y_j \phi_j(t, \alpha) = 0. \end{array} \quad (13)$$

Then we have

$$\Phi(t, \alpha, \lambda) := \left[\begin{array}{l} G(K)\phi(t, \alpha) + \alpha - \phi(t, \alpha) - \lambda y - e \\ y^T \phi(t, \alpha) \end{array} \right] = 0, \quad (14)$$

where $\Phi : \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{n+1}$.

Let $z = (t, \alpha, \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ and define $\Theta :\in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{n+2}$,

$$\Theta(z) = \Theta(t, \alpha, \lambda) = \left[\begin{array}{c} t \\ \Phi(z) \end{array} \right]. \quad (15)$$

It can be seen that Θ is continuously differentiable at any $z \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$.

Proposition 1. *For any $z = (t, \alpha, \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ with $t > 0$, the matrix $\Theta'(z)$ is nonsingular.*

Proof. Since $C > 0$, there exist relative interior points for problem (4).

$$\Theta'(t, \alpha, \lambda) = \left[\begin{array}{ccc} 1 & 0 & 0 \\ (G(K) + I)\phi'_t(t, \alpha) & M(z) & -y \\ y^T \phi\prime_t(t, \alpha) & y^T \phi'_\alpha(t, \alpha) & 0 \end{array} \right] \quad (16)$$

where

$$M(z) = G(K)\phi'_\alpha(t, \alpha) + I - \phi'_\alpha(t, \alpha), \quad (17)$$

and $\phi'_t : \mathbb{R}^{n+1} \to \mathbb{R}^n$ is the gradient of ϕ with respect to t. $\phi'_\alpha(t, \alpha) = \mathrm{diag}\{q_i\}$ is a $n \times n$ diagonal matrix with its elements

$$q_i(z) = \frac{1}{2}[(\frac{\alpha_i}{\sqrt{\alpha_i^2 + 4t^2}}) + (\frac{(C - \alpha_i)}{\sqrt{(C - \alpha_i)^2 + 4t^2}})], \quad i = 1, 2, \cdots, n. \quad (18)$$

Let $dz := (dt, d\alpha, d\lambda)^T \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ satisfies $\Theta'(z)dz = 0$. Then from the above equations $dt = 0$ and

$$M(z)d\alpha - yd\lambda = 0, \quad (19)$$

$$y^T \phi'_\alpha(t, \alpha) d\alpha = 0. \tag{20}$$

Since $\phi'_\alpha(t, \alpha)$ is positive definite for $t > 0$ then we have from (20) that $y^T d\alpha = 0$. Substitute it into (19) and left multiply (19) by $(d\alpha)^T$ we have

$$(d\alpha)^T M(z)(d\alpha) + (d\alpha)^T y d\lambda = 0,$$

which means that $d\alpha = 0$. Since $G(K)$ is positive semidefinite and y_i takes value at $\{-1, 1\}$, from (19) we have that $d\lambda = 0$. Therefore, $\Theta'(z)$ is nonsingular for any z with $t > 0$.

In smoothing Newton method, people usually take $\|\Theta(z)\|^2$ as the merit function. However, since the data mining problems usually produce large-scale optimization problems, in order to overcome this difficulty in numerical computation, we introduce the following merit function $\Upsilon(z)$:

$$\Upsilon(z) = \sum_{j=1}^{n} \rho_{h_j}(\Theta_j(z))$$

where

$$\rho_{h_j}(\xi) = \begin{cases} \xi^2/2, & \text{if } |\xi| \le h_j, \\ h_j|\xi| - h_j^2/2, & \text{otherwise} \end{cases}$$

is the well-known Huber function [7]. Since Huber function is linear when ξ is not small enough, the computation on the merit function will be very simple when the iteration is far from the optimal solution. However, we can not simply apply Newton's method on $\Upsilon(z)$ since Huber function is smooth but not second order differentiable. Before we give the smoothing Newton method, we let $\gamma \in (0, 1)$ is a real number and define $\beta : \mathbb{R}^{n+2} \to \mathbb{R}_+$,

$$\beta(z) := \gamma \min\{1, ||\Theta(z)||\},$$

and

$$\Omega := \{z = (t, \alpha, \lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R} |\ u \ge \beta(z)\bar{t}\},$$

where $\bar{t}$ is a given positive number. Now we are ready to give the smoothing Newton method.

Algorithm 1

Step 0. Choose constants $\bar{t} > 0, \delta \in (0, 1)$ and $\sigma \in (0, 1/2)$. Let $\bar{z} = (\bar{t}, 0, 0) \in \mathbb{R}_+ \times \mathbb{R}^n \times \mathbb{R}$, $t^0 = \bar{t}$ and $(\alpha^0, \lambda^0) \in \mathbb{R}^{n+1}$ be an arbitrary point. Set $l := 0$.

Step 1. If $\Theta(z^l) = 0$ then stop. Otherwise, let $\beta_l := \beta(z^l)$.

Step 2. Compute $\Delta z^l := (\Delta t^l, \Delta\alpha^l, \Delta\lambda) \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}$ by

$$\Theta(z^l) + \Theta'(z^l)\Delta z^l = \beta_l \bar{z}. \tag{21}$$

Step 3. Let m_l be the smallest nonnegative integer m satisfying

$$\Upsilon(z^l + \delta^m \Delta z^l) \le [1 - 2\sigma(1 - \gamma\|\bar{t}\|)\delta^m]\Upsilon(z^l). \tag{22}$$

Define $z^{l+1} := z^l + \delta^{m_l} \Delta z^l$.

Step 4. Set $l := l + 1$ and go to Step 1.

From Proposition 1 and by using nonsmooth analysis, we can prove the following convergence theorem of the above algorithm. We omit its proof here because of the page limitation.

Theorem 1. *Suppose that z^* is an accumulation point of the infinite sequence $\{z^k\}$ generated by Algorithm 1. If all $V \in \partial\Theta(z^*)$ are nonsingular, then for any initial point $z^0 = (t^0, \alpha^0, \lambda^0)$ with $t^0 > 0$, the sequence $\{z^k\}$ converges to z^* quadratically. i.,e., for k large enough,*

$$\|z^{k+1} - z^*\| = O(\|z^k - z^*\|^2)$$

and

$$t^{k+1} = O(t^k)^2.$$

3 Numerical Tests

In this section we report our numerical results of smoothing Newton methods for data classification of the heart disease, ionosphere and breast-cancer-Wisconsin data obtained from the UCI repository. Our numerical experiments were carried out in personal computer with 1060GHz AMD Sempron (tm) Processor and 512MB memory and 80G hard disk. The program is written in MATLAB 7.0.1.

3.1 The Pre-process of the Data

In data mining, the raw data we got may include very big and/or very small numbers. As we known, in numerical computation by computer, the big number may 'eat' the small number. In order to overcome this difficulty and obtain good computation results from Algorithm 1, we need to do the data pre-process first, which includes normalization and standardization for the data.

The data normalization and standardization we took are as follows,

$$v^{'} = v - min(v) \tag{23}$$

$$v^{''} = v^{'}/max(v^{'}) \tag{24}$$

where the vector v is a $n \times 1$ vector, and it presents an attribute of the data, (23) presents the standardization of the data and (24) presents the normalization of the data. When there is an attribute which has the same number in every sample, that is v is a vector of the same number, $v^{'}$ will be zero vector, and $v^{''}$ will be not feasible. Thus we must delete the attributes which have the same numbers in every sample, so that the revised data can be feasible.

3.2 Numerical Results

Now we present the numerical results of smoothing Newton method for the soft margin data classification problem on the heart disease, ionosphere, breast-cancer-Wisconsin, and the data of Credit Cards. For the convenience of expression, we denote the linear kernel function $k_1(x_1, x_2) = x_1.x_2{}'$ as K_1, polynomial

kernel function $k_2(x_1, x_2) = (x_1.x_2{}' + 1)^d$ as K_2 , and the Gaussian kernel function $k_3(x_1, x_2) = exp(-(x_1 - x_2)^T(x_1 - x_2)/\sigma^2)$ as K_3.

We choose the values of all parameters in Algorithm 1 as

$$\gamma = 0.01, \quad \sigma = 0.25, \quad \delta = 0.5, \quad \epsilon = 0.0001, \quad C = 1.$$

For each data-set, we randomly take 60% of the data as the training set and the left 40% as the test set. Numerical results on standard benchmark datasets are summarized in Table 1, Table 2 and Table 4. In the process of computation, numerical differentiation can be used to take the place of accurate derivative. We take two-point differentiation formula in our numerical tests:

$$f'(x) \approx \frac{f(x+\epsilon) - f(x)}{\epsilon}, \qquad \epsilon > 0. \tag{25}$$

We first apply Algorithm 1 on the training set and obtain the value α, then we use it to classify the test data to check the accuracy of our methods. In Table 1 below, we listed the processing time (in seconds) and the number of iteration for working out the value of α. For example, in the heart disease data-set, the total number of the data is 270, the computation time is 4.969 seconds, and the number of Newton iterations in Algorithm 1 is 18 for using the linear kernel function K_1.

Table 1. Time and number of iterations in computation

	total	K_1	K_2	K_3	K_3
			$d = 2$	$\sigma^2 = 1$	$\sigma^2 = 0.2$
heart disease	270	4.969 18	6.203 22	4.156 15	5.047 17
ionosphere	351	12.328 25	56.188 111	7.75 15	7.094 14
breast-cancer-W	699	100.64 47	59.000 29	39.734 19	41.719 19

After we obtained the value of α from Algorithm 1, we can easily check the accuracy of our algorithm by applying it on the test set, and it only takes little time. Table 2 below listed the accuracy of our algorithm with different kernel functions.

Table 2. The accuracy of Algorithm 1

	K_1	K_2	K_3	K_3
		$d = 2$	$\sigma^2 = 1$	$\sigma^2 = 0.2$
heart disease	86.36%	77.27%	77.27%	78.18%
ionosphere	92.20%	96.45%	97.87%	78.01%
breast-cancer-W	98.57%	98.92%	98.57%	98.21%

In Table 3 below we listed the accuracy of data classification by using semi-definite programming (SDP) on the l_1 soft margin classification problem with $C = 1$ in [12]. It can be seen from Table 2 and Table 3 that Algorithm 1 can provide better classification for the problems except the polynomial kernel classification K_2 with $d = 2$ for heart disease data.

Table 3. Accuracy of SDP in [12]

	K_1	K_2	K_3	K_3
		$d = 2$	$\sigma^2 = 1$	$\sigma^2 = 0.2$
heart disease	84.3%	79.3%	59.5%	–
ionosphere	83.1%	94.5%	92.1%	–
breast-cancer-W	87.7%	96.4%	89.0%	–

Table 4. Test results for credit cards data

Kernal	Parametes	Time (min)	Iterations	Accuracy
K_1		5.4083	57	79.39%
K_2	$d = 2$	3.8000	31	83.26%
K_3	$\sigma^2 = 2$	1.5516	16	84.19%
K_3	$\sigma^2 = 1$	1.5576	16	84.15%

Finally, we apply Algorithm on the data of Credit Cards. There are 6000 samples in total, 5040 of which are good credit, 960 of which are bad credit. Each sample involves 65 attributes, which were processed into numbers. Because the number of the samples is large, the data was randomly partitioned into 10% training and 90% test sets, that is the training data-set has 600 samples(500 good, 100 bad samples). The Table 4 below shows the detail of Algorithm 1 on the problem.

4 Conclusions

In the paper, we provide a smoothing Newton method for support vector machine model for l_1 soft magian data classification problem. The algorithm is global and local quadratic convergent. Numerical tests on some well-know data classification problem shows that the method is fast and can provider better accuracy that the method in literature. Further research on variety of kernel matrix classification and smoothing Newton methods for ν-support vector machines [2] and multi-class classification [15] are under going.

Acknowledgments. This research has been partially supported by a grant from National Natural Science Foundation of China (#10671203, #70621001, #70531040, #70501030, #10601064, #70472074) and Faculty Research Grant of Minnesota State University Mankato.

References

1. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines. Cambridge University Press, Cambridge (2000)
2. Chen, P., Lin, C., Schölkopf, B.: A tutorial on ν-support vector machines. Appl. Stoch. Models Bus. Ind. 21, 111–136 (2005)

3. Chen, X., Qi, L., Sun, D.: Global and superlinear convergence of smoothing Newton method and its application to general box constrasined variational inequalities. Math. Comp. 67, 519–540 (1998)
4. Cortes, C., Vapnik, V.: Support vector networks. Machine Learning 20, 1–25 (1995)
5. Ferris, M.C., Munson, T.S.: Semismooth support vector machines. Math. Program. Ser.B 101, 185–204 (2004)
6. Harker, P.T., Pang, J.-S.: Finite-dimensional variational inequality and nonlinear complementarity problem: A survey of theory, algorithm ans applications. Math. Program. 48, 161–220 (1990)
7. Huber, P.J.: Robust regression: Asymptotics, conjectures, and Monte Carlo: Ann. Statist. 1, 799–821 (1973)
8. Qi, L., Sun, D., Zhou, G.: A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities. Math. program. 87, 1–37 (2000)
9. Mangasarian, O.L.: Mathematical programming in data mining. Data Mining and Knowledge Discovery 1, 183–201 (1997)
10. Mangasarian, O.L., Musicant, D.R.: Lagrangian support vector machines. Journal of Machine Learning Research 1, 161–177 (2001)
11. Mangasarian, O.L.: A finite Newton method for classification. Optim. Methods Softw. 17, 913–929 (2002)
12. Lanckriet, G.R.G., Cristianini, N., Ghaoui, L.E., Bartlett, P., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. J. Machine Learning Research 5, 27–72 (2004)
13. Schölkopf, B., Smola, A.J.: Learning with Kernels– Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press, Cambridge (2002)
14. Vapnik, V.N.: The Nature of Statistical Learning Theory, 2nd edn. Springer, Heidelberg (2000)
15. Zhong, P., Fukushima, M.: Regularized nonsmooth Newton method for multi-class support vector machines. Optim. Methods Softw. 22, 225–236 (2007)

Short-Term Capital Flows in China: Trend, Determinants and Policy Implications

Haizhen Yang[1,2], Yanping Zhao[1], and Yujing Ze[1]

[1] Graduate University of Chinese Academy of Sciences,
Beijing 100190, China
[2] Research Centre on Fictitious Economy and Data Science, CAS,
Beijing 100190, China
haizheny@gucas.ac.cn

Abstract. The volatility of international capital flows have further increased both in volume and speed since the outbreak of subprime crisis originating from America. Orientation of international capital flows blurred because of the downward expectation on the growth rate in main countries. Since the short-term capital flow has gradually become an important part of international capital flow in China in decade, the volatility of short-term capital flows may affect the development of Chinese economy severely. A structural model-VECM was build to explore the determinants of net flows of short-term capital in China. The conclusions of this study were that net flows in China are largely determined by estate price, circulated stock value, expectation on exchange rate and interest rate. On that basis, some policy suggestions were proposed.

Keywords: short-term capital inflow, determinants, VECM.

1 Introduction

As the development of financial integration, the volatility of international capital increased both in volume and speed, which should be viewed as a mixed blessing. On one hand, the international capital flows have brought about advanced management, technology and growth potential; and on the other hand, a surge in capital flows will probably bring about some difficulties on monitoring and threat the development of economy. The financial crises have always been related to periodic international capital flows and fluctuations. Specifically, short-term international capital, with high fluctuations, may mess economy quickly. This is due to its variability, as the capital coming in quickly can flow out just as fast.

Generally speaking, a loan or investment within a period less than a year is defined as "short-term capital", while over a year is known as "long-term capital". However, with the development of financial market and the innovation of financial instruments, the line between "short-term capital" and "long-term capital" has blurred. In view of the standards adopted by Balance of Payment and the relatively strict regulation on international capital flows in China, we identify short-term capital as a combination of portfolio investments, other investment flows (except long-term loans), and the partly un-tracked implicit capital represented by errors and omissions.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 552–560, 2009.

Figure 1 shows both the absolute amount of capital inflows and outflows has surged in decade, the ratios of short-term inflows to the total private inflows have greatly increased while the ratios of short-term outflows to the total private outflows are stable. CNY has appreciated gradually accompanying with a boom in housing and stock market since the exchange rate reform in 2005. In the mean while, short-term capital flow to China is on a large scale. Inflows of short-term capital raised inflation pressure, which could influence the steady development of China.

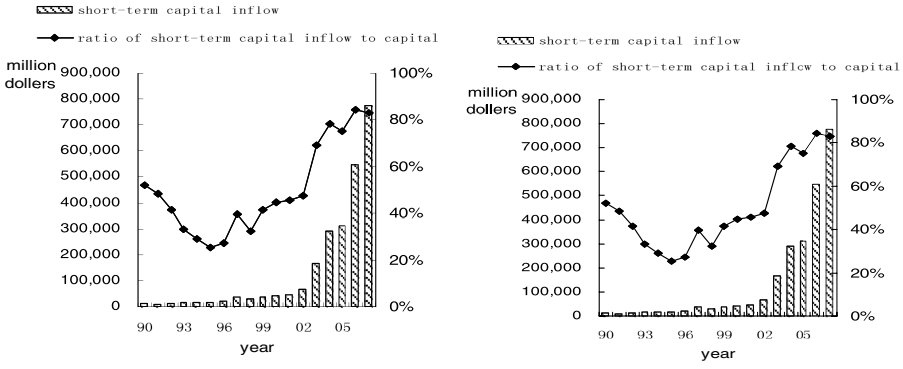


Fig. 1. Short-term capital in China

China's economy is facing new challenges in the global economy crises. Short-term capital flow to China is under a pressure of outflow. In this circumstance, on the basis of summarize the driving causes of net flows of short-term capital; empirical study the determinants of capital flows in China in decade has significance.

2 Literature Review

Some studies focusing on the determinants of short-term capital flows have been achieved. Most scholars classified the factors affecting capital flows into two categories, which are "pull factors" reflecting domestic opportunity and risk, and "push factors" reflecting global change and influence. Schadler et al. [1] showed that pull factors are more important. But Chuhan et al. [2] argued that compared with domestic factors, global factors, such as a fall in US interest rates and the slow down in US production were more important or at least as important. Chuhan et al further concluded that lower international interest could explain more than 50% of the capital flowing to emerging countries. He also pointed out that compared with bond flows; equity flows were more responsive to global factors; bond flows were more responsive to a country's credit rating and the secondary-market price of debt. In-Mee Baek [3] observed that portfolio investment in Asia was dominantly pushed by investors' appetite for risk and other external factors, while in Latin American, it was pulled mainly by rapid economic growth and global factors rather than the market's risk appetite.

Since China's accession to the WTO, the studies focus on the determinants of short-term capital flows in China was prevalent (Wang [4], Wang [5], Liu [6]). Jiang [7] identified arbitrage as the primary determinant of short-term capital flows. She also suggested that both financial innovation and development of capital markets made transactional activities remarkably convenient. Wang(2003)[8], Liu and Wen [9] analyzed the determining factors, such as interest rate, exchange rate, and assets price employing OLS (Ordinary Least Squares)., Zhang, Pei, and Fang [10] analyzed inflows of short-term international capital employing a triple-arbitrage model. Lan and Chen [11] counted the co-integration relationship among interest rate, exchange rate expectation, and capital control, which indicated that the above-mentioned variables are highly correlated.

Previous studies have made great progress in analyzing the determinants of capital flows. However, there are still some topics need further discussions. First of all, there are few studies concerning new features of short-term capital flows surge in China after Southeastern Asian financial crisis. In addition, most studies applied multiple regression analysis, which can not reflect dynamic progress perfectly. This paper will use dynamic econometrics methods to identify the driving factors affecting short-term capital flows in China. What's more, quarterly data, rather than annual data, have been adopted to enhance the frequency of empirical testing.

3 Determinants of Short-Term Capital Flows

Based on Mundell-Fleming Model, which is the extension of Interest Rate Parity Theory, risk reward of arbitrage capital flows are defined as: $p = r_d - r_f - \Delta E$

Where

P denotes risk reward pursued by arbitrage capital flows,

r_d denotes domestic interest rates,

r_d denotes foreign interest rates,

ΔE denotes static expected exchange rate.

The model indicates that the risk reward can be divided into two parts. One part is interest difference, short-term capital always flow from countries with lower interest to higher ones. Another part is reward from expected fluctuations in the exchange rate. The mechanism is that the expectation on currency appreciation brings about short-term capital inflows; correspondingly, the expectation on currency depreciation brings about outflows. In view of Mundell-Fleming Model and costs of short-term capital flows in China is high, we must consider real interest difference, which have adjusted for inflation. The fluctuations of exchange rate may be another important driving factor because the expectation on CNY's appreciation has occurred since 2002, which may induce capital flow.

Short-term capital flows may aim at the revenue on the appreciation of real assets and revenue on portfolio investment in considerate of China is undergoing the economic transform. Real estate price and circulated stock value listed on the Shenzhen and Shanghai stock markets are adopted in this study as "push factors" proxies.

A dummy variable of policy is added in the model. China carried out reform on currency exchange rate and established a managed floating exchange rate regime in July 21st, 2005. Henceforth, the fixed exchange rate regime became flexible and marketization of CNY exchange rate mechanism has been speeded up.

4 Model-Building and Empirical Results

A quarterly model was build to examine the main driving factors in view of the complexity of the short-term capital flows. Vector Error Correction Model has been chosen to ensure almost short-term fluctuations to be covered in. Furthermore, mechanisms of reclamation from short-term non-equilibrium to long-term equilibrium were analyzed.

4.1 Data Specification

The interval for each variable is from the first quarter in 1999 to the fourth quarter in 2007. The data on net flows of short-term capital (SCF) is taken from Balance of Payment for China in million-dollars, and the calculation formula for SCF follows the former definition of short-term capital flows in section 1 (each item uses credit side of figures).

The real interest difference between China and America is signified as RR. Three-month deposit rate which originals from Reuters is substituted for the nominal interest rate in China, while the nominal interest rate in America is signified as yield on three-month Treasury bill on secondary market comes from the web of Federal Reserve. The CPI in China comes from macro-economic warning system of People's Bank of China. The CPI in America comes from the web of U.S. Department of Labor. RR equals to the interest rate difference between China and America minus the CPI difference between China and America. Revenue on expected exchange rate is signified as DE, $DE = -\frac{E^e - E_0}{E_0}$ NDF is a proxy of E^e, which comes from Reuters; E_0 is spot exchange rate, which comes from the web of The University of British Columbia. Circulated stock value abbreviated as CSV, comes from database in China Economic Information Network. The values at the end of each season are selected, and then adjust to dollars according to the average exchange rate in the month. RE is housing price index also comes from database in China economic information network. The dummy variable of policy is signified as 0 before the second quarter in 2005(included) and 1 after the third quarter in 2005. D (SCF) is first difference value of SCF, and more.

4.2 Tests for Stationarity and Johansen Co-integration Analysis

If two or more time series are non-stationary, but a linear combination of them is stable, then the series are said to be co-integrated. The co-integration can be taken as kind of long-run equilibrium. The co-integrated variables may fluctuate in the short run, but in a long run, they can regress to their intrinsic relationship.

A test for stationarity must be implemented to assure the time series are stationarity of the same differences before the co-integration analysis, in short, I (d). The ADF test shows that all time series in this study are I (1) at 1% significant level.

We employ the Johansen co-integration analysis to test SCF and other time series to identify the long-term relationships.

Table 1. Johansen co-integration analysis

variable	level			first difference		
	(C, T, L)	ADF value	P value	(C, T, L)	ADF value	P value
SCF	(c,t,0)	-3.9568	0.0119	(c,t,2)	-6.0457	0.0001
RR	(c,0,0)	-2.9873	0.0459	(c,0,1)	-8.4857	0.0000
DE	(c,0,0)	-1.2395	0.6461	(c,0,1)	-7.1014	0.0000
CSV	(c,0,3)	-1.8599	0.3461	(c,0,0)	-3.4645	0.0154
RE	(c,0,0)	-0.9346	0.7651	(c,0,0)	-4.9965	0.0003

Table 1 shows a co-integration equation can be deduced at 1% significant level. After normalized co-integrating coefficients, the equation is as follows:

$$SCF = \underset{[-6.1679]}{5323.499}\,RR + \underset{[-2.41835]}{11528.29}\,DE - \underset{[6.31861]}{22068.03}\,CSV + \underset{[-3.26743]}{2514.099}\,RE + 7987.563$$

The long-term relationships between short-term capital flows and the influence factors are shown in equation. T-tests are all significant. The results indicate that, RR is in line with theoretical expectations, the symbol before RR is plus, high rewards of domestic interest may attract arbitrage, and in the meanwhile, net flows of short-term capital increase. The symbol before DE, which signifies the expected exchange rate, is also plus, can explain the capital flows due to expected appreciation. The plus before RE shows that the net flows of short-term capital may increase along with ascend of the housing price. The minus before CSV shows the negative relationship between net flows and circulated stock value, which indicates the investors abroad mainly pay attention to the appreciation on real asset.

4.3 Empirical Results

The co-integration can be taken as kind of the long-run equilibrium. On the basis of co-integration vector, VECM model for SCF is as follows:

$$vecm_{t-1} = SCF_{t-1} - 5323.499\,RR_{t-1} + 11528.29\,DE_{t-1} + 22068.03\,CSV_{t-1} - 2514.099\,RE_{t-1} - 7987.563$$

Table 2. VECM for SCFI

Error correction	$\Delta SCFI_t$	Error correction	$\Delta SCFI_t$
$vecm_{t-1}$	-1.010483 [-2.35390]	ΔDE_{t-2}	-30492.35 [-1.99344]
$\Delta SCFI_{t-1}$	0.103105 [0.33340]	ΔCSV_{t-1}	106197.5 [2.22259]
$\Delta SCFI_{t-2}$	0.302734 [1.14520]	ΔCSV_{t-2}	152.4894 [0.00363]
ΔRR_{t-1}	-4271.186 [-1.68191]	ΔRE_{t-1}	-3389.338 [-0.73199]
ΔRR_{t-2}	-3393.388 [-1.20147]	ΔRE_{t-2}	576.9738 [0.13439]
ΔDE_{t-1}	-10977.46 [-0.81861]	$Policy$	-36034.19 [-1.99784]
C	5067.679 [0.80402]		

Note:△ denote first difference, [ ] is t value

Table 2 shows VECM model for SCF. Although some individual T-tests are not significant due to insufficient data, excessive variables and weak degree of freedom, the overall results show that the selected variables are appropriate. The error correction coefficient of SCF is -1.010483, with a significant t-value. The error correction mechanism can be illustrated in the following way: In t-1 period, if $VECM_{t-1} > 0$, it implies that the net flows of short-term capital in that period are more than long-run equilibrium level. Due to a negative adjustment mechanism, when D (SCF) diminished, the net flows reduced in t period, the net flows regress to long-run equilibrium level. If $VECM_{t-1} < 0$, adjustment can be similarly deduced in opposite.

4.4 Impulse Response and Variance Decomposition

Impulse response and variance decomposition are used to examine the dynamic relationships among the variables and analyze the influences.

Figure 2 shows the responses of SCF when affected by other factors. When affected by a positive S.D fluctuation of RR, the fluctuation will have a positive effect on SCF in the first phase, and a trivial negative effect in the second phase, then a positive effect less than that in the first stage, finally gradually steady, which imply the interest difference between America and China will induce net flows of short-term capital in the beginning, and the effect will diminish gradually in the long term. DE signifies revenue on expected exchange will also have a positive affect on SCF in the beginning, and then the signs alternate positive and negative, finally stable. A positive S.D fluctuation of CSV will induce net flows, and

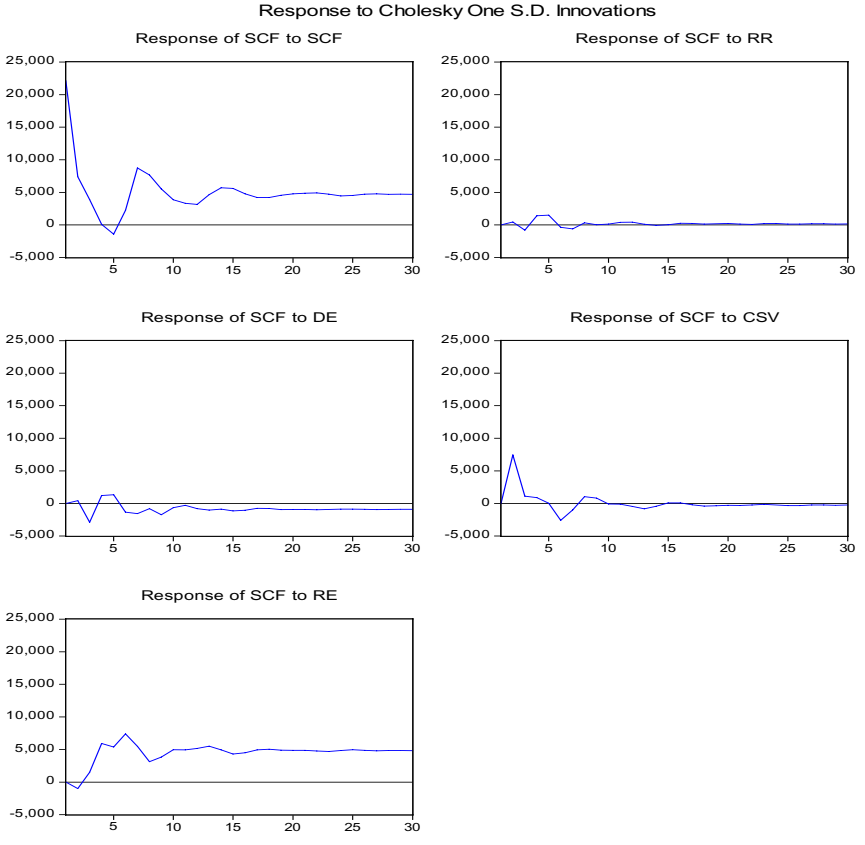


Fig. 2. Impulse Response of SCF

the effect will diminish gradually. When attacked by a positive affect by RE, the ascending assets price will induce capital outflow, then a larger outflow, finally stable.

Decomposition of variance can further analyze the dynamic character in the model.

It is a method that decomposes the variance of endogenous fluctuations into parts. The results are shown in table 3. The fluctuations of estate price account for about 27% of the S.D fluctuations in SCF; interest difference account for about 25%; circulated stock value account for about 12%; expected exchange rate account for about 6%; itself account for others, which can be explained with inertia effect and demonstration effect. In summarize, estate price, circulated stock value, interest difference, and expected exchange rate are all factors affecting capital flows in China.

Table 3. Variance Decomposition

Period	S.E.	SCF	RR	DE	CSV	RE
1	22155.9	44.77054	39.75263	3.615741	11.70573	0.155356
2	24541.76	36.80186	36.53518	3.089613	23.25038	0.322965
3	25094.45	35.81934	35.46722	5.082509	23.01345	0.617487
4	25862.51	33.76188	33.58362	5.068184	21.71653	5.869789
5	26527.05	32.28554	31.93097	5.277863	20.7145	9.791136
6	27783.95	30.6112	29.26959	5.226298	19.21132	15.68159
7	29696.29	31.74517	28.48274	5.795558	17.30931	16.66721
8	30852.93	31.53333	29.07664	5.869653	17.22657	16.29381
9	31632.74	30.99378	28.88132	6.329609	16.96647	16.82882
10	32256.89	30.51549	28.38028	6.262848	16.44116	18.40022
11	32801.26	29.99504	27.97288	6.122794	15.97241	19.93688
12	33367.79	29.42716	27.51086	6.074669	15.45371	21.5336
13	34160.15	29.15266	27.02668	6.094267	14.78852	22.93788
14	34995.06	29.17387	26.77683	6.116053	14.2732	23.66005
15	35718.43	29.07358	26.6942	6.23961	13.96468	24.02793
16	36329.85	28.8075	26.57677	6.302436	13.6708	24.64249
17	36909.69	28.52537	26.33351	6.266014	13.33292	25.54218
18	37496.01	28.30165	26.0506	6.236245	12.98428	26.42723
19	38100.13	28.11432	25.84793	6.251238	12.65604	27.13047
20	38714.22	27.94418	25.69906	6.264335	12.35328	27.73914

Cholesky Ordering: RR DE CSV RE SCF

5 Conclusions

Summing up the results of the empirical test, and taking the China's short-term capital movement characteristics and financial system into consideration, we conclude as follows:

First of all, net flows of short-term capital are affected by factors such as real interest difference, expected exchange rate, and estate price, which are all labile**,** rather than fundamental economic factors. We once introduced variable GDP to reflect the fundamental economic (including fundamental facility construction, the quality and quantity of labors, and market prospects) in the model, which induce a remarkable ascending in AIC and SC. The result indicated that short-term capital in China is accompanied with arbitrage and speculation.

Secondly, the fluctuations of stock market and housing are the most important determinants in net flows of short-term capital. The fluctuations of stock market are determinants, which reflect the arbitrage intention of short-term capital flows. But the

relationship between stock market and net flows is complex. Once the stock market achieved high enough, the risk increased, and short-term capital may flow out, which has indicated as a negative relationship in co-integration equation. The capital flows are also affected by the estate price, which have also shown in impulse response and variance decomposition. The fluctuations of estate price account for about 27% of the S.D fluctuations in SCF, which indicated the capital flows speculating on housing. These flows helped in pushing the ascending of housing in China.

Thirdly, the expectation on the CNY appreciation is an important factor affecting net flows of short-term capital. The impact of expected exchange rate on net flows is about 25% in variance decomposition. There is a stable long term positive relationship between the expected exchange rate and capital flows. The effect of expected exchange rate is only about 6% in VECM, which is not large enough. But in considerate of the pressure from real interest difference, the effect may be larger.

In general, short-term capital flow mechanisms are complex. There has been a surge in capital flow to China in order to speculation on interest, exchange rate, securities and asset price. In this circumstance, strict supervision and control on the fluctuations of exchange rate, as well as improvements in the CNY exchange rate regime are in need. The government should also strength risk management in stock market; stabilize housing market; further improve monitor on short-term capital flow; take precautions against financial risks promptly; properly liberalize capital account prudently.

Acknowledgement

This work is funded by NSFC (grant No.:70673100, 70621001) and Graduate University of Chinese Academy of Sciences.

References

1. Schadler, S., Carkovic, M., Bennett, A., Khan, R.: Recent Experiences with Surges in Capital Inflows, IMF Occasional Paper no. 108, Washington DC (1993)
2. Chuhan, P., Claessens, S., Mamingi, N.: Equity and Bond Flows to Asia and Latin America. IMF Working Paper 1160 (1993)
3. Baek, I.-M.: Portfolio investment flows to Asia and Latin America: pull, push or market sentiment. Journal of Asian Economics 17, 363–373 (2006)
4. Wang, Y.: Research on China's capital flow since1994. International Financial Research 6, 67–73 (2004)
5. Wang, Q.: Econometric studies on determinants of capital movement in China. International Financial Research 6, 64–69 (2006)
6. Liu, L.D.: Research on capital inflows in China. Financial Research 3, 62–70 (2007)
7. Jiang, L.Q.: Cause, infection, and prevention of international short-term capital flows. Shanghai Financial Transaction 4, 5–6 (1997)
8. Wang, X.: Cause of short-term capital inflows. International Financial Research 1, 59–64 (2003)
9. Liu, H.H., Wen, T.: Short-term capital inflows: Causes and policy proposal. Zhongnan University of Economics and Law Transaction 6, 122–130 (2005)
10. Zhang, Y.H., Pei, P., Fang, X.M.: Short-term capital inflows and triple- arbitrage model based on interest rate, exchange rate and assets price. International Financial Research 9, 41–52 (2007)
11. Lan, Z.H., Chen, L.: The scale of Short-term capital flows and Econometric studies. Financial Economic 4, 48–49 (2007)

Finding the Hidden Pattern of Credit Card Holder's Churn: A Case of China

Guangli Nie[1], Guoxun Wang[1,3], Peng Zhang[1], Yingjie Tian[1], and Yong Shi[1,2,*]

[1] Research Center on Fictitious Economy and Data Science,
CAS, Beijing 100190, China
[2] College of Information Science and Technology, University of Nebraska at Omaha,
Omaha, NE 68182, USA
[3] School of Computer science and information engineering, Henan University,
Kaifeng 475001, China
sdungl@163.com, wangguoxun06@mails.gucas.ac.cn,
zhangpeng04@gmail.com, tianyingjie1213@163.com, yshi@gucas.ac.cn

Abstract. In this paper, we propose a framework of the whole process of churn prediction of credit card holder. In order to make the knowledge extracted from data mining more executable, we take the execution of the model into account during the whole process from variable designing to model understanding. Using the Logistic regression, we build a model based on the data of more than 5000 credit card holders. The tests of model perform very well.

Keywords: credit card churn, data mining, business intelligence.

1 Introduction

With cost-cutting and intensive competitive pressure, nowadays more and more companies start to focus on Customer Relationship Management (CRM). The unknown future behaviors of the customers are quite important to CRM. It is of crucial importance to detect the customers' future decision then the company can take corresponding actions early [6]. The customers who stop using the products or services of a company are usually called churners. Finding the churners can help the companies to retain the customers. Gustafsson, Johnson, and Roos (2005) studied telecommunication services to examine the effects of customer satisfaction and behavior on customer retention [7]. Results indicated a need for CRM managers to determine customer satisfaction more accurately in order to reduce customer churn.

One of the major reasons for this is that it cost less to retain existing customers than to acquire new customers [15]. It costs up to five times as much to make a sale to a new customer as it does to make an additional sale to an existing customer[4][17]. It is thought that 20% of a company's customers can account for 80% of its business [11]. And, it is becoming more evident that the only way to retain customers who bring profit in this industry is not only to be customer-driven but also to focus on building long-term relationships.

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 561–569, 2009.

Due to the development of information technology, many companies have accumulated a large amount of data. Analyzing the data stored in the database can help the managers make the right marketing decision and pinpoint the right customers to market.

It is a very good field to predict the churn of credit card holders as the data of credit cards is more accessible and the churn of credit card holders is serious. Several studies have proved the effectiveness of the power of customer retention in credit cards. Van den Poel and Lariviere (2004) calculated the financial impacts of one percent increase in customer retention rate [18]. A bank is able to increase its profits by 85% due to a 5% improvement in the retention rate [14].

The main purpose of this paper is to provide a framework of understanding the knowledge of the card holders' hidden pattern instead of providing a new data mining algorithm, focusing on the application of the churn prediction. From data preparation to useful knowledge, the application goal of churn prediction covers the whole process of this paper. In this paper, we introduce a way to execute churn prediction considering execution and effectiveness of the model.

The rest of the paper is organized as follows. The definition of churn is introduced in Section 2. Section 3 summarizes the algorithms building model and criteria evaluating model. The data used in the research is described in Section 4, and the modeling process is presented in Section 5. The conclusions are introduced in the last section.

2 Definition of Churn

Churn is defined differently in different fields. Churn refers to the customer shift from one service provider to another [10]. Many of the existing researches use the behaviors related to product and a threshold fixed by a business rule to define churn. Once the transactions of the customer is low than the threshold, the customer would be regarded as a churner [6]. Van den Poel and Larivi`ere (2004) defined a churner as someone who closed his accounts [18]. Buckinx and Van den Poel (2005) defined a partial defector as someone with the frequency of purchases below the average and the ratio of the standard deviation of the interpurchase time to the mean interpurchase time above the average [1]. Glady, N.(2008) defined a churner as a customer with less than 2500 Euros of assets (savings, securities or other kinds of products) at the bank[6].Scott A. Neslin regarded customer as a churn according to the propensity of customers to cease doing business with a company in a given time period [13].

Churn in the specific telecommunications industry is also a hot topic. The broad definition of churn is the action that a customer's telecommunications service is canceled which includes both service-provider initiated churn such as customer's account being closed because of payment default and customer initiated churn. In the telecommunication study, only customer initiated churn is considered and it is defined by a series of cancel reason codes. Examples of reason codes are: unacceptable call quality, more favorable competitor's pricing plan, misinformation given by sales, customer expectation not met, billing problem, moving, change in business, and so on [10].

The relationship between churn rate and average lifetime is also studied. If no new customers are acquired then the average lifetime of an existing customer is equal to 1 / c, where c is the annual churn rate [13]. The study of Gustafsson, Johnson, and Roos (2005) includes customer satisfaction (CSt), affective commitment (ACt), calculative

commitment (CCt), a situational trigger condition (STt), and a reactional trigger condition (RTt), all in time t, to predict churn in time t + 1 ($Churn_{t+1}$)[7].

In our application, we define that the customer who do not do any transaction with the bank on his own initiative during the observation period (explained later) is a churner.

3 Algorithms and Evaluation Criteria

In order to predict the churn of the customer effectively, it is of crucial importance to build an effective model which fulfills evaluation criteria. To accomplish this, there exist a lot of predictive modeling techniques available. Researchers use a variety of "approaches" to develop churn models, described by a combination of estimation technique, variable selection procedure, time allocations to various steps in the model-building process, and a number of variables included in the model [13]. These data mining algorithms can help to select variables and build model [9]. The techniques include GA, Regression, Neural Networks, Decision Tree, Markov Model, Cluster Analysis [8].

According to the research of John Hadden [8], regression and decision tree are the two most popular algorithms used in the research and perform well. Neslin categorized the approaches as "Logit," "Trees," "Novice," "Discriminant," and "Explain." After comparison they found that the Logit and Tree approaches perform the best and result in firm predictive ability, the Novice approach is associated with middle-of-the road predictive performance, while the Discriminant and Explain approaches are associated with lower predictive performance [13].

In this paper, we also use these two algorithms to predict the churn of credit card holder and give the threshold of the early-warning alert system.

After building a predictive model, marketers want to use these classification models to predict future behavior of the customers. It is essential to evaluate the performance of the classifier. PCC, ROC are usually used as criteria. PCC, also known as accuracy, is undoubtedly the most commonly used evaluation metric of a classifier. PCC computes the ratio of correctly classified cases to the total number of cases to be classified. The receiver operating curve (ROC) is a graphical plot of the sensitivity which is the number of true positives versus the total number of events and 1-specificity which is the number of true negatives versus the total number of non-events. The ROC can also be represented by plotting the fraction of true positives versus the fraction of false positives [3]. We will evaluate the model by these criterias.

4 Research Data

4.1 Data Source

A major China commercial bank provided the data for this study. The data is extracted from a data warehouse of that bank. All of the data is organized at the level of customer. No matter in which branch the customer open the account, the data warehouse can identify the customer by name and identification number of the customer. All of the customers in the warehouse are indexed by a unique customer number. The data

warehouse records all the past change of the card. Take *the balance of the card* for example, once the balance of the card changed, there will be one more row to keep the new balance and the last balance is still kept.

The data warehouse is built in 2005, thus the data can track back to Jan 2005. There are 60 million customers in the data warehouse. We sample randomly from the system and the time interval of data is from Jan 2005 through the end of performance period (i.e. censoring on April 30, 2008) or the end of the customer relationship (i.e. churn). This is only raw data and we will refine the data when we calculate the variables.

The data relates to all the aspects of the credit card holder including the personal information of the card holder, the basic information of the card, the detailed transaction information, the abnormal usage information of the card and so on. There are 9 tables related to credit card and the detailed tables are list in following table.

Table 1. The tables related to the credit card

No.	Name	Description
1	Daily balance of the card	The daily balance record of the credit card
2	Information of the card	The basic information of the credit card
3	Table of the abnormal usage	The record about abnormal usage of the card
4	Table of limit usage rate	The rate of credit limit utilization when first time used
5	Table of first time used	The information about the first time use of the card
6	Table of revoking pay	The information about the bank revokes the pay of card
7	Table of suspending pay	The information the holders do not pay off on time
8	Table of the transactions of the cards	The information of the transactions of the holder
9	Personal information	The detailed information of the card holder

Dealing with the time in right way is quite important in this research. The time interval used in a telecommunication is 9-month. This period make sure that there is more variation using the 9-month cumulative churn measure [7]. Bart Larivi`ere(2004) dealt with the time by customer's lifecycle. We can see that dealing with the time issue properly is quite important in this churn application [12].

The final aim of the churn study is to predict what will happen to the customer in the future, thus we must split the time window into two phases i.e. the observation period and the performance period. In the observation period, we design variables to observe the transaction behaviors of a customer, and then we check whether the customer becomes a churner or not in the performance period. When we build model the independent variables(X_1 X_2 ……) are calculated from the data happened during the observation period and the dependent variable(Y) are calculated from the data happened during the performance period. In order to avoid the affect of the festival or

season, we set a whole year (12 months) as observation period (the interval between t_1 and $t_{2)}$ and a whole year (12 months) as performance period (the interval between t_2 and t_3). In our study the observation period is from Jan 1, 06 to Dec 1, 2006 and the performance period is from Jan 1 2007 to Dec 31, 2007. After the observation of a whole year, our model will predict whether the customer become a churner or not in the performance period. The observation period must be a whole one, so Customer C of Fig. 1 will not be used.

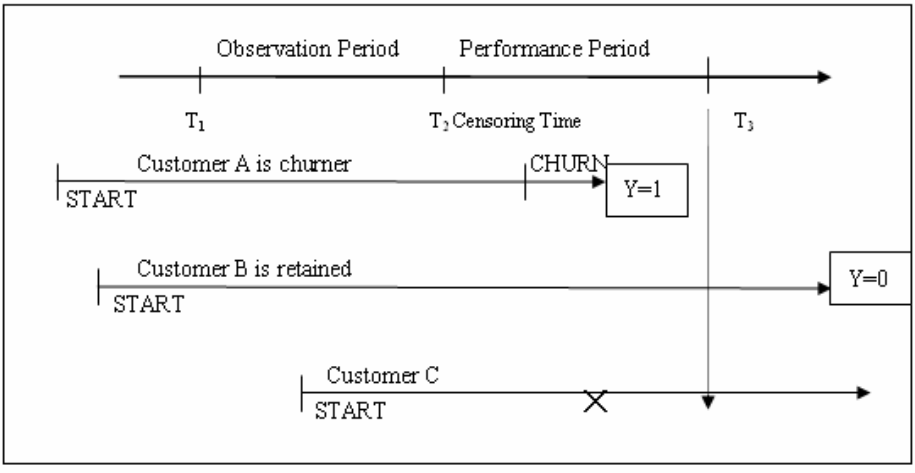


Fig. 1. The time window of analysis

4.2 Variables Design

Different types of the data may contribute differently to the model. The good derivable variables can help to build good model. Future researches could also study the types of data that are important for churn prediction models. For example, while the data includes behavioral, customer interaction, and demographic variables, it contains very little data which is directly useful for marketing. In different application, the input data varies. In a research of wireless industry, the input data are categorized in to four types: Demographics, Usage level, Quality of Service (QOS), Features/Marketing paging [17].

We do not give any assumption before data mining. We would not assume that some factors would affect the dependent variable while others do not affect in advance. The task of this phase is to design as many variables as possible. We designed the derivative variables from six perspectives: demographic variables, frequency variables, the total amount variables, time related variables, extreme variables and the status variables.

4.3 Data Description

The raw data we study consists of 9 tables and the largest table has 8 million records. The population consists of churner (without any activities during the performance

period) and retained customers. All transactions are aggregated at the customer level. One customer may have several cards, but the object to predict is the customer, thus it is quite necessary to aggregate the data at the customer level. There are many methods to aggregate the data to customer level. One of the functions of the extreme variables is to aggregate the records to one as mentioned.

After the data preparation, we get a data mart consisting of 5456 samples, 440 are churners (8.1%) and 5016 do not churn (91.9%). Before building models, we separate the sample into training set to build the classifiers and a test set for the performance assessment.

According to the variable designed in Section 4.2, we calculate the derivative variables. There are 172 variables reflecting the complete information of the customer. The independent variables are calculated from the data during the observation period from January 2006 till December 2006 (12 months). The dependent variable (Y) is calculated from the data during the performance period from January 2007 till December 2007 (12 months). According the definition of Section 2, we define that the customer who did not do any transaction with the bank on his own initiative during the observation period is a churner.

$$Y = \begin{cases} 1 \text{ if customer has no transaction in performance period} \\ 0 \text{ if customer has at least one transaction in performance period} \end{cases} \quad (1)$$

5 Modeling and Explanation

The goal of this research is to show how to predict the churn of predict holder in an easy to execute and easy to understand way. Novel algorithm is not the aim of this research. The classifier applied in this research is well-known data mining algorithms: a logistic regression. This algorithm is quite mature and widely used. A lot of researches have proved the high accuracy and effectiveness of the algorithm [6] [8] [13].

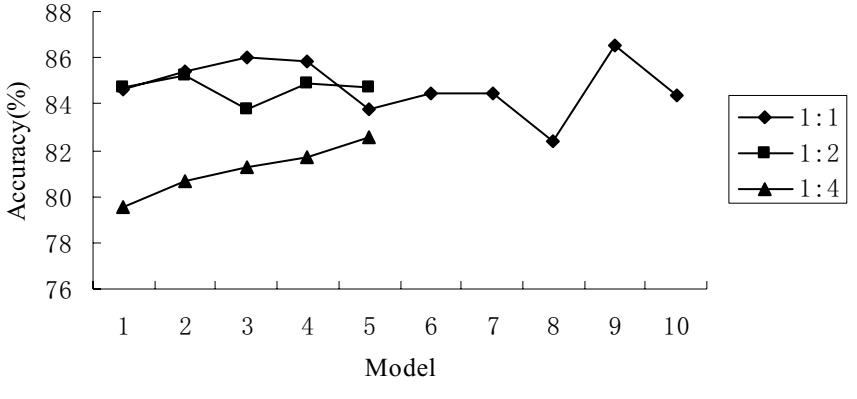


Fig. 2. The accuracy of the model with different proportion

From last section, we can see that the proportion between current customers and churner of sample data is around 1: 9, which is unbalanced. In order to get the suitable proportion of the samples to build model, we try three proportions: 1:1, 1:2 and 1:4. The results of the proportions are list in the following Fig. 2. We can see that the proportion 1:1 performs the best and 1:4 perform worst. This means that the proportion 1:1 is the most suitable to existing data.

So we build ten models at the proportion 1:1 randomly sampling, which perform best. For the static churn model, a split sample design is used. Fifty percent of the churner (220 samples) and five percent of the non-churner (270 samples) is used to train the model; all of the rest (about 4960) is used to validate the model.

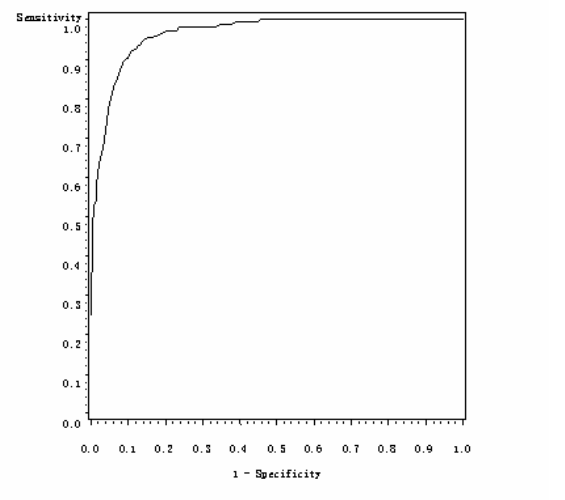


Fig. 3. The ROC curve of the classifier

The following rule reflects the hidden pattern of the credit card holder

m=-10.087+0.00705*The proportion of overdraft time out of total time +1.6182 *The maximum length without action+1.0594*Whether the accountant being abnormal+3.6168*The times risk level higher than 3 recent 9 months+(-6.91E-07)*The number of cards callback+(-3.9418)*The length between issued and open of the cards+(-0.00229)*The number of new cards in recent year+0.00573*The length between open and used for the first time of the cards+0.035*The length from last transaction of loan to now+(-0.0814)*The times expired of free interest+3.117*The loan times reason for deposit+(-1.4932)*The proportion of the trade amount during National day out the whole month+(-10.087)*Whether VIP;

```
P=1+exp(m);
 If P>0.5
    then the customer will churn in the next year;
Else
    the customer will not churn in the next year;
end.
```

As introduced in the last section, the data mart consists of 172 variables. We used stepwise to reduce the variables. There are 13 variables chosen to build the model. The accuracy shown in Fig. 2 is the result of validation. We chose the best model as the final one. The ROC curve of the chosen model is shown in Fig. 3. The ROC curve shows that classifier performs very well.

The decision can be made based on above rule. Some of the variables are constant information such as the variable *Whether VIP*. This variable can only be used to predict the churn of the customer.

6 Conclusions

In this research, we propose the whole process of churn prediction of credit card and we demonstrate the whole process using a case of China. The purpose of this research is not to propose a new algorithm. The focus of this study is the execution and the understanding of the model. The suitable design of derivable variables and the right way to build model could be helpful to execute of the rule.

We split the time window into observation period and performance period. The independent variables are calculated from the data during the observation period, and the dependent variables are calculated from the data during the performance period.

After the data preparation, we get a data mart consisting of 5456 samples, 440 are churners(8.1%) and 5016 do not churn (91.9%). Based on the well-known data mining algorithms logistic regression, we build a very stable and accurate model. The variables are quite easy to conduct since we take the execution in account during the whole process.

Acknowledgements

The authors are very grateful to the anonymous bank that supplied the data to finish the analysis. This research has been partially supported by a grant from National Natural Science Foundation of China (#70621001, #70531040, #70501030, #70472074), Beijing Natural Science Foundation (#9073020), 973 Project #2004CB720103, Ministry of Science and Technology, China, and BHP Billiton Co., Australia.

References

1. Buckinx, W., Van den Poel, D.: Customer base analysis: Partial defection of behaviorally-loyal clients in a non-contractual fmcg retail setting. European Journal of Operational Research 164(1), 252–268 (2005)
2. Chiang, D.-A., Wang, Y.-F., Lee, S.-L., Lin, C.-J.: Goal-oriented sequential pattern for network banking churn analysis. Expert Systems with Applications 25(3), 293–302 (2003)
3. Coussement, K., van de Poel, D.V.: Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. Expert Systems with Applications 34, 313–327 (2008)

4. Dixon, M.: 39 Experts Predict the Future. America's Community Banker 8(7), 20–31 (1999)
5. Floyd, T.: Creating a New Customer Experience. Bank Systems and Technology 37(1), R8–R13 (2000)
6. Glady, N., Baesens, B., Croux, C.: Modeling Churn Using Customer Lifetime Value. European Journal of Operational Research (2008), doi:10.1016/j.ejor.2008.06.027
7. Gustafsson, A., Johnson, M.D., Roos, I.: The effects of customer satisfaction, relationship commitment dimensions, and triggers on customer retention. Journal of Marketing 69(4), 210–218 (2005)
8. Hadden, J., Tiwari, A., et al.: Computer assisted customer churn management:State-of-the-art and future trends. Computers & Operations Research 34, 2902–2917 (2005)
9. Hung, S.-Y., Yen, D.C., Wang, H.-Y.: Applying data mining to telecom churn management. Expert Systems with Applications 31(3), 515–524 (2006)
10. Lu, J.: Predicting Customer Churn in the Telecommunications Industry -— An Application of Survival Analysis Modeling Using SAS. Sprint Communications Company
11. Kotler, P.: Marketing Management. Prentice-Hall, NJ (2000)
12. Lariviere, B., Van den Poel, D.: Investigating the role of product features in preventing customer churn, by using survival analysis and choice modeling: the case of financial services. Expert Systems with Applications 27, 277–285 (2004)
13. Neslin, S.A., Gupta, S., et al.: Defection Detection: Improving Predictive Accuracy of Customer Churn Models (2004)
14. Reichheld, F.F., Sasser Jr., W.E.: Zero defections: quality comes to service. Harvard Business Review 68(5), 105–111 (1990)
15. Roberts, J.H.: Developing New Rules for New Markets. Journal of the Academy of Marketing Science 28(1), 31–44 (2000)
16. Roos, I.: Switching Processes in Customer Relationships. Journal of Service Research 2, 76–93 (1999)
17. Slater, S.F., Narver, J.C.: Intelligence Generation and Superior Customer Value. Journal of the Academy of Marketing Science 28(1), 120–127 (2000)
18. Van den Poel, D., Larivi`ere, B.: Customer attrition analysis for financial services using proportional hazard models. European Journal of Operational Research 157(1), 196–217 (2004)
19. Zhao, Y., Li, B., et al.: Customer Churn Prediction Using Improved One-Class Support Vector Machine ADMA (2005)

Nearest Neighbor Convex Hull Classification Method for Face Recognition

Xiaofei Zhou[1,*] and Yong Shi[1,2]

[1] Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences, Beijing 100190, China
[2] College of Information Science and Technology University of Nebraska at Omaha, Omaha, NE 68182, USA
zhouxf@gucas.ac.cn,
yshi@gucas.ac.cn, yshi@unomaha.edu

Abstract. In this paper, nearest neighbor convex hull (NNCH) classification approach is used for face recognition. In NNCH classifier, a convex hull of training samples of a class is taken as the distribution estimation of the class, and Euclidean distance from a test sample to the convex hull (the distance is called convex hull distance) is taken as the similarity measure for classification. Experiments on face data show that the nearest neighbor convex hull approach can lead to better results than those of 1-nearest neighbor (1-NN) classifier and SVM classifiers.

Keywords: classification, SVM, convex, nearest neighbor convex hull, face recognition.

1 Introduction

As one of most important biometrics technologies, face recognition has become an active research in pattern recognition and data mining area. Similar to many pattern recognition problems, solving classification problem is a key for face recognition. During the past 20 years many classification methods have been successfully applied in face recognition, such as 1-NN [1,2], Neural Networks [3], SVM [4], HMM [5] etc. As a non-parametric pattern recognition approach, the single nearest neighbor (1-NN) classifier is often used for classification. 1-NN is a most intuitive approach based on the nearest neighbor rule, which decides a query to the class including the nearest prototype to it. However, the performance of 1-NN is limited by the available prototypes in each class, which depends on how prototypes are chosen to account for possible sample variations and also how many prototypes are available. Practically no matter how representative the prototypes may be, there are always un-prototyped viewings, because only a finite, often small, number of prototypes are available as compared to all possibilities [6]. To adapt for more prototypes changes than the original prototypes, many classification approaches have been presented by using the linear combinations

* Corresponding Author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 570–577, 2009.

of the prototypes to expand the representational capacity of them, such as NLC [6], NFL [7], NFP and NFS [8], NNL and NNP [9] etc. Some classifiers also utilize convex hull of prototypes to represent training set. In [10], k-local convex distance is used as measure, and k-local convex hulls of each class are used to represent the class. Different from the above method, Jiang et al. [11] and Zhou et al. [12] present a new classification idea called nearest neighbor convex hull (NNCH) classification, which utilizes convex hull of all prototypes per class to represent each class. NCH method [13] mentioned by Nalbantov et al. has similar ideas with NNCH. The approaches mentioned above are all based on the nearest neighbor rule and provide an infinite number of prototypical points to represent the query. In this paper, we apply the NNCH method [11,12] for face recognition. Like 1-NN, NNCH is a non-parameters method, which uses convex hull of prototypes to represent each class, and Euclidean distance between the query and the convex hull is used as the measure for nearest neighbor classification. The experiments on Yale face database and FERET face database show good performance.

The rest of the paper is organized as follows: Section 2 introduces foundations of NNCH, Section 3 describes NNCH method, Section 4 presents experimental results on face recognition.

2 Foundations of NNCH

Definition 1 [Convex Set]. Let $\boldsymbol{G} \subseteq \boldsymbol{R}^d$. Say that $\boldsymbol{G}$ is convex set if, for each $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in $\boldsymbol{G}$, the line segment between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is contained in $\boldsymbol{G}$: $\forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \boldsymbol{G}$ and $\lambda \in [0,1]$, $\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 \in \boldsymbol{G}$.

That is, $\boldsymbol{G}$ is convex if, for all $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in $\boldsymbol{G}$ and $\lambda \in [0,1]$, the point $\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2$ lies in $\boldsymbol{G}$.

Given a set $\boldsymbol{S}$ with n elements $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$, where $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{id})^T$ is d dimensional vector in a feature space. We use $\mathrm{co}(\boldsymbol{S})$ to denote the convex hull of $\boldsymbol{S}$, $\mathrm{co}(\boldsymbol{S})$ is the smallest convex set containing set $\boldsymbol{S}$:

Definition 2 [Convex Hull]. The convex hull of a set $\boldsymbol{S} \subset \boldsymbol{R}^d$ is the smallest convex set containing set $\boldsymbol{S}$: $\mathrm{co}(\boldsymbol{S}) = \{\sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i \mid \boldsymbol{x}_i \in \boldsymbol{S}, \alpha_i \geq 0, \sum_{i=1}^{n} \alpha_i = 1, i = 1,2,\cdots n\}$.

The convex hull of a set $\boldsymbol{S}$ is simply the set of all linear combinations of elements of $\boldsymbol{S}$ in which the coefficients of elements of $\boldsymbol{S}$ are nonnegative and sum to 1. Such constrained linear combinations are known as convex combinations.

In this paper the distance measure based on convex hull is used for classification, i.e., the distance from a query to a convex hull of training samples of a class is taken as the similarity measure. Virtually, the distance problem is a projection problem. Given a nonempty set $\boldsymbol{G} \subset \boldsymbol{R}^d$ and a vector $\boldsymbol{y}$, the projection problem is the problem of determining the point $\hat{\boldsymbol{x}} \in \boldsymbol{G}$ that is the closest to $\boldsymbol{y}$ among all $\boldsymbol{x} \in \boldsymbol{G}$ (with respect to the Euclidean distance). Formally, the problem is given by

$$\min_{x \in G} \|y - x\|^2 \tag{1}$$

When the set G is a closed and convex set, the solution exists and it is unique, as seen in the following theorem.

Theorem 1 [Projection onto Closed Convex Set]. Let H is a Hilbert space, and $G \subset H$ be a nonempty closed convex set and $y \in H$.

- (Existence and uniqueness) There is a unique $\hat{x} \in G$ such that $\hat{x} = \arg\min_{x \in G} \|y - x\|$. $\hat{x}$ is called the projection of y onto G and is denoted by $p_G(y)$.
- (Characterization) A point $\hat{x} \in S$ is the projection $p_G(y)$ if $< y - \hat{x}, x - \hat{x} > \leq 0$ for all $x \in S$.

The theorem guarantees the existence and uniqueness of the projection of a vector on a closed convex set. That is, for a nonempty closed convex set $G \subset H$ and arbitrary $y \in H$, $d(y, G) = \min_{x \in G} \|y - x\|$, where $d(y, G)$ can be computed, and there is a unique $\hat{x} \in G$ such that $\|y - \hat{x}\| = \min_{x \in G} \|y - x\|$. The projection $\hat{x}$ is the convex combination that is nearest to the query y . The distance between the query and nearest point in the convex hull is used as the similarity of the query and the class set. The unique solution y to the projection problem is referred to as the projection of $\hat{x}$ on $co(S)$.

3 Nearest Neighbor Convex Hull Classifier

Inspired by both nearest neighbor rule and the geometric interpretation of SVM, Jiang et al. [11] and Zhou et al. [12] presented a new classification method: nearest neighbor convex hull (NNCH) method. They use convex hull to extend the representation of prototypes and adopt nearest neighbor rule to realize classification.

The idea of NNCH is to expand representational capacity of prototypes of each class by convex combinations. This virtually provides an infinite number of prototypical points, and thus can account for more prototypical changes than the original prototypes. In the calculation of distance between a query vector and a class, the query is projected to the convex hull spanned by the prototypes of this class. The projection point is the convex combination that is nearest to the query. The distance between the query and convex hull is used as the basis for classification. Based on such distances, the conventional 1-NN classification, which compares each prototype individually, is extended to the nearest neighbor convex hull classification, which compares the convex hulls of each class: the query is classified to the class of the nearest convex hull.

The idea using convex hull of all samples in a class to represent the class is also derived from the geometric interpretation of SVM. SVM is a robust methodology [14, 15] for classification. Intuitively, given a set of points belonging to two classes, SVM finds the optimal hyperplane that separates binary class data without errors, while maximizing the distance from either class to the hyperplane. In geometry, this is equivalent to separating the convex hulls of all samples in each class with maximal margin [16, 17]. So the

convex hull of all samples in a class can represent the class space of samples well. In NNCH, such convex hulls are considered to extend the representation of prototypes. But unlike maximal margin rule of SVM, NNCH uses the nearest neighbor rule to classify. In fact, NNCH can be considered as an approach separating the convex hull by nearest neighbor rule. In reference [10], k-local convex distances are used for classification, which emphasis the local linear construction of sample distribution and use local convex hulls to represent the class. But in NNCH, general convex hull of a class is presented to represent sample distribution, which is inspired by the convex hull in SVM.

3.1 Convex Hull Distance

In NNCH classifier, a convex hull of training samples of a class is taken as distribution estimation of the class, and Euclidean distance from a test sample to the convex hull (the distance is called convex hull distance) is taken as the similarity measure for classification.

Given a set $\boldsymbol{S} \subset \boldsymbol{R}^d$, $\boldsymbol{S} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k\}$, the distance function between a query $\boldsymbol{x}$ and the convex hull of $\boldsymbol{S}$ can be written in detail:

$$\begin{aligned} d^2(\boldsymbol{x}, co(\boldsymbol{S})) &= \min_{\boldsymbol{\eta} \in co(\boldsymbol{S})} \|\boldsymbol{x} - \boldsymbol{\eta}\|^2 \\ &= \min_{\boldsymbol{\alpha}} \left\| \boldsymbol{x} - \sum_{i=1}^{k} \alpha_i \boldsymbol{x}_i \right\|^2 \\ &= \min_{\boldsymbol{\alpha}} [(\boldsymbol{x} \cdot \boldsymbol{x}) - 2\sum_{i=1}^{k} \alpha_i (\boldsymbol{x} \cdot \boldsymbol{x}_i) + \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j (\boldsymbol{x}_i \cdot \boldsymbol{x}_j)] \end{aligned} \tag{2}$$

$$\text{s.t. } \sum_{i=1}^{k} \alpha_i = 1,\ \alpha_i \geq 0,\ i = 1,2,\cdots,k$$

Let $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_k)$, $\boldsymbol{e} = (1,1,\cdots,1)^T_{1\times k}$, $\boldsymbol{0} = (0,0,\cdots,0)^T_{1\times k}$, $\boldsymbol{\eta} = \boldsymbol{X}\boldsymbol{\alpha}$, $\boldsymbol{e}^T\boldsymbol{\alpha} = 1, \boldsymbol{\alpha} \geq \boldsymbol{0}$. Equation (2) can be written in matrix:

$$\begin{aligned} d^2(\boldsymbol{x}, co(\boldsymbol{S})) &= \min_{\boldsymbol{y} \in co(\boldsymbol{S})} \|\boldsymbol{x} - \boldsymbol{\eta}\|^2 \\ &= \min_{\boldsymbol{\alpha}} \|\boldsymbol{x} - \boldsymbol{X}\boldsymbol{\alpha}\|^2 \\ &= \min_{\boldsymbol{\alpha}} (\boldsymbol{x}^T\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{X}\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\alpha}) \end{aligned} \tag{3}$$

$$\text{s.t.} \quad \boldsymbol{e}^T\boldsymbol{\alpha} = 1, \boldsymbol{\alpha} \geq \boldsymbol{0}$$

As $\boldsymbol{x}^T\boldsymbol{x}$ is constant, which can be discard from the optimal problem. The optimal problem we need to solve is

$$\begin{aligned} &\min_{\boldsymbol{\alpha}} \ 2\boldsymbol{x}^T\boldsymbol{X}\boldsymbol{\alpha} + \boldsymbol{\alpha}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\alpha} \\ &\text{s.t.} \quad \boldsymbol{e}^T\boldsymbol{\alpha} = 1,\ \boldsymbol{\alpha} \geq \boldsymbol{0} \end{aligned} \tag{4}$$

Equation (4) is a convex quadratic optimization. In this paper, we use the MATLAB optimal tools to realize the solution.

Supposed $\boldsymbol{\alpha}^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_k^*)^T$ is the optimum solution for (4), projection $\boldsymbol{y}$ is the convex combination of elements in $\boldsymbol{S}$, with the coefficient $\alpha_1^*, \alpha_2^*, \cdots, \alpha_k^*$:

$$\boldsymbol{y}=\sum_{i=1}^{k}\alpha_i^*\boldsymbol{x}_i \tag{5}$$

Then we can compute $d^2(\boldsymbol{x},co(\boldsymbol{S}))$ by $\boldsymbol{y}$:

$$d^2(\boldsymbol{x},co(\boldsymbol{S}))=\left\|\boldsymbol{x}-\boldsymbol{y}\right\|^2=\left\|\boldsymbol{x}-\sum_{i=1}^{k}\alpha_i^*\boldsymbol{x}_i\right\|^2 . \tag{6}$$

3.2 Nearest Neighbor Convex Hull Algorithm

We assume to have a multi-labeled training set $\boldsymbol{S}=\{(\boldsymbol{x}_1,c_1),...,(\boldsymbol{x}_n,c_n)\}$, where $\boldsymbol{x}_i\in\boldsymbol{R}^d$ is a sample and $c_i\in\{1,...,l\}$ is its corresponding class or label, i.e. there are l categories training sets: $\boldsymbol{S}_1=\{\boldsymbol{x}_i|c_i=1\}$, $\boldsymbol{S}_2=\{\boldsymbol{x}_i|c_i=2\}$, … , $\boldsymbol{S}_l=\{\boldsymbol{x}_i|c_i=l\}$. The l convex hulls from different category training sets are:

$$co(\boldsymbol{S}_1)=\left\{\sum_{i,ci=1}\alpha_i\boldsymbol{x}_i\;\middle|\;\sum_{i,ci=1}\alpha_i=1,\ \alpha_i\geq 0\right\},$$

$$\cdots,$$

$$co(\boldsymbol{S}_l)=\left\{\sum_{i,ci=l}\alpha_i\boldsymbol{x}_i\;\middle|\;\sum_{i,ci=l}\alpha_i=1,\ \alpha_i\geq 0\right\}.$$

For an arbitrary query $(\boldsymbol{x},c)$, $\boldsymbol{x}\subset\mathbf{R}^d$, c is class label which is unknown, we need to give the c value. We respectively compute the square distance between $\boldsymbol{x}$ and each class convex hull:

$$d^2(\boldsymbol{x},co(\boldsymbol{S}_1))=\min_{\boldsymbol{\eta}1\in co(\boldsymbol{S}1)}\left\|\boldsymbol{x}-\boldsymbol{\eta}_1\right\|^2,$$

$$\cdots,$$

$$d^2(\boldsymbol{x},co(\boldsymbol{S}_l))=\min_{\boldsymbol{\eta}l\in co(\boldsymbol{S}l)}\left\|\boldsymbol{x}-\boldsymbol{\eta}_l\right\|^2.$$

We take $d^2(\boldsymbol{x},co(\boldsymbol{S}_j))$ as the similarity of $\boldsymbol{x}$ and the jth class, and classify $\boldsymbol{x}$ to the class of the nearest neighbor convex hull:

$$c=\arg\min_j d^2(\boldsymbol{x},co(\boldsymbol{S}_j)),\ j=1,2,\cdots,l . \tag{7}$$

Compared with 1-NN, NNCH presents infinite samples convex combined by training samples, whereas 1-NN has the limited training set. Compared with SVM, NNCH can be directly used for multi-class problems. For SVM, it usually needs to decompose multi-class problems into many two-class problems.

4 Experiments

We apply NNCH classifier on face recognition. The comparison experiments of NNCH with 1-NN and SVM are conducted on two face databases, ORL and FERET face

databases. For SVM, three kernels, linear kernel $k = (\boldsymbol{x} \cdot \boldsymbol{y})$, quadratic kernel $k = (1 + (\boldsymbol{x} \cdot \boldsymbol{y}))^2$ and gaussian kernel $k=exp\ (-0.5(\| \boldsymbol{x}\text{-}\boldsymbol{y} \| /\sigma)^2)$ are chosen, and a bottom-up binary tree-structured approach [4] is used to extend SVM to deal with multi-class problems. All of our experiments are carried out under Matlab 7.0 platform.

The first experiment is performed on ORL face database (http://www.cl.cam.ac.uk/Research/DTG/attarchive/facedatabase.html). ORL database contains 40 distinct persons. Each person has ten different images labeled with the number 1,2, … ,10. The original face images were all sized 92×112 pixels with 256-level gray scale. Fig.1 shows ten face images of one subject. In our experiments, all images are transformed to JPEG format, and downsampled to 16×16 by bicubic interpolation. We use the first two images of each individual and their mirrors as training set. The remaining 320 images are as test set.

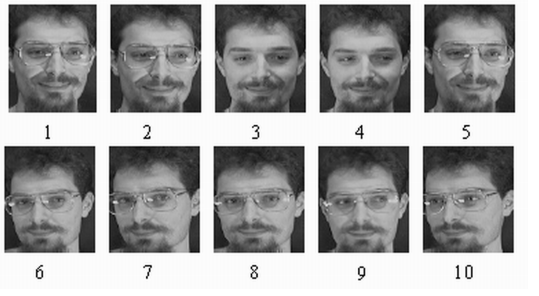


Fig. 1. Ten face images of one person in ORL face database

Table 1 gives the comparisons of different methods on face recognition accuracy. The penalty parameter of SVMs adopt C=∞, and for gaussian kernel SVM, σ=3.

Table 1. Results of experiments on ORL face database

Classification methods	Recognition accuracy
1-NN	86.56%
SVM(Linear kernel)	85.63%
SVM(quad kernel)	83.00%
SVM(gaussian kernel)	86.88 %
NNCH	87.81%

The second experiment is performed on a subset of FERET face database (http://www.itl.nist.go/iad/humanid/feret /feret_master.html). The subset contains 200 distinct persons. Each person has seven different images labeled with"ba, bd, be, bf, bg, bj, bk". The original face gray images are cropped to 80×80 pixels. Fig.2 shows the cropped face images from one subject.

Fig. 2. Seven face images for one person in FERET face database

In our experiments, the images are also downsampled to 40×40 by bicubic interpolation, and the illumination normalization technique is done for compensating illumination variations of all faces. The first four images (labeled with"ba, bd, be, bf") of each person and their mirrors are used for training, and the rest 600 images are used for testing.

Table 2 gives the comparisons of different methods on face recognition accuracy. The penalty parameter for SVMs adopt $C=\infty$, and for gaussian kernel SVM, $\sigma=50$.

Table 2. Results of experiments on FERET face database

Classification methods	Recognition accuracy
1-NN	75.00%
SVM(linear kernel)	81.50%
SVM(quadratic kernel)	83.00%
SVM(gaussian kernel)	83.67%
NNCH	86.50%

From the results of above experiments, it is clear that as a whole the face classification performance of the NNCH is competitive with that of 1-NN and SVMs. In Table 1 and Table 2, NNCH on face recognition all can obtain the highest recognition accuracy among the methods. In Table 1, the recognition rate of NNCH (87.81%) is higher than 1-NN (86.56%) and three kernel SVMs (85.63%, 83.00% and 86.88 %); In Table2, the recognition rate of NNCH is 86.50%, which is much higher 11.5% than 1-NN, also higher 5% than linear SVM, and higher almost 3% than nonlinear SVMs.

5 Conclusions

This paper introduces a novel pattern classification method called nearest neighbor convex hull (NNCH) approach for face recognition. In NNCH, a convex combination of vectors belonging to a class is used to define a measure of distance from a query vector to the class. The measure is defined as the Euclidean distance from the query to the nearest convex combination. Experiments on face data show that NNCH leads to better results than those of 1-nearest neighbor (1-NN) classifier and SVM classifiers.

Acknowledgments. This work was partially supported by the National Grand Fundamental Research 973 Program of China under Grant No.2004CB720103, by the National Nature Science Foundation of China under Grant No.70531040, No.70621001, No.10601064 and No.70501030 and by a research grant from BHP Billion Co., Australia.

References

1. Jin, Z., Yang, J.Y., Hu, Z.S., Lou, Z.: Face Recognition based on Uncorrelated Discriminant Transformation. Pattern Recognition 34(7), 1405–1416 (2001)
2. Yang, J.Y., Zhang, David, Yang, J.Y., et al.: Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. IEEE PAMI 26, 1131–1137 (2004)
3. Ranganath, S., Arun, K.: Face Recognition Using Transform Features and Neural Networks. Pattern Recognition 30(10), 1615–1622 (1997)
4. Guo, G., Li, S.Z., Chan, K.L.: Face Recognition by Support Vector Machines. In: Proc. fourth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 196–201 (2000)
5. Samaria, F., Young, S.: HMM based Architecture for Face Recognition. Image and Computer Vision 12, 537–543 (1994)
6. Li, S.Z.: Face Recognition based on Nearest Linear Combinations. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 839–844 (1998)
7. Li, S.Z.: Performance Evaluation of the Nearest Feature Line Method in Image Classification and Retrieval. IEEE Trans. On PAMI 22(11), 1335–1339 (2000)
8. Chen, J.T., Wu, C.C.: Discriminant Waveletfaces and Nearest Feature Classifiers for Face Recognition. IEEE Trans. On PAMI 24(12), 1644–1649 (2002)
9. Zheng, W.M., Zou, C., Zhao, L.: Face Recognition Using Two Novel Nearest Neighbor Classifiers. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 5, pp. v-725–728 (2004)
10. Vincent, P., Bengio, Y.: K-local Hyperplane and Convex Distance Nearest Neighbor Algorithms. In: Dietterich, T.G., BeckerS, G.Z. (eds.) Advances in Neural Information Processing Systems 14, pp. 985–992. MIT Press, Cambridge (2002)
11. Jiang, W.H., Zhou, X.F., Yang, J.Y.: P-norm Nearest Neighbor Convex Hull Classification Algorithms (in Chinese). Journal of Harbin Institute of Technology 38, 982–984 (2006); In: Proceedings of the 7th Chinese Symposium on Intelligent Robot(2006)
12. Zhou, X.F., Jiang, W.H., Yang, J.Y.: Kernel Nearest Neighbor Convex Hull Classification Algorithm (in Chinese). Journal of Image and Graphics, 1209–1213 (2007)
13. Nalbantov, G.I., Groenen, P.J.F., Bioch, J.C.: Nearest Convex Hull Classification, Report (2007), http://repub.eur.nl/publications/index/728919268/NCH10.pdf
14. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, New York (1995)
15. Deng, N.Y., Tian, Y.J.: A New Approach in Data Mine-Support Vector Machine. Science Press, Beijing (2004)
16. Bennett, K.P., Bredensteiner, E.J.: Duality and Geometry in SVM Classifiers. In: Langley, P. (ed.) Proceedings of the 17th International Conference on Machine Learning, pp. 57–64. Morgan Kaufmann, San Francisco (2000)
17. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K.: A Fast Iterative Nearest Point Algorithm for Support Vector Machine Classifier Design. IEEE Transactions on Neural Networks 11(1), 124–136 (2000)

The Measurement of Distinguishing Ability of Classification in Data Mining Model and Its Statistical Significance

Lingling Zhang[1,2,*], Qingxi Wang[1], Jie Wei[1], Xiao Wang[1], and Yong Shi[2,3]

[1] Graduate University of Chinese Academy of Sciences, Beijing (100190), China
[2] Research Centre on Fictitious Economy and Data Science, CAS, Beijing (100190), China
[3] College of Information Science and Technology, University of Nebraska at Omaha, Omaha, NE 68118, USA
Tel.: 86-10-82680676
{zhangll,qingxiwang,Jiewei,xiaowang,yshi}@gucas.ac.cn

Abstract. In order to test to what extent can data mining distinguish from observation points of different types, the indicators that can measure the difference between the distribution of positive and negative point scores are raised. First of all, we use the overlapping area of two types of point distributions-overlapping degree, to describe the difference, and discuss the nature of overlapping degree. Secondly, we put forward the image and quantitative indicators with the ability to distinguish different models: Lorenz curve, Gini coefficient, AR, as well as the similar ROC curve and AUC. We have proved AUC and AR are completely linear related; Finally, we construct the nonparametric statistics of AUC, however, the difference of K-S is that we cannot draw the conclusion that zero assumption is more difficult to be rejected when negative points take up a smaller proportion.

Keywords: Data Mining Model, Distinguishing Ability Measurement.

1 Introduction

As for the analysis of the results of data mining and knowledge assimilation, the usual practice is presenting the results of data mining as a visual form to users by systems, and then re-analysising and re-classifying them subjectively by users providing amendment and reference for the next data mining, so as to assure the entire data mining system of more robust and higher availability[1]. Compared with the traditional analysis of data (such as query, reporting, analysis of the on-line applications), the essential difference between data mining and them is that data mining means digging information and discovering knowledge without clear assumption[2]. In other words, on the one hand the results of data mining which is a means of artificial intelligence has the "subjective" characteristic owned by systems, but on the other hand there isn't absolutely correct pre-analysis which can be used as the basis for

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 578–587, 2009.

comparison with the results of data mining and evaluating the quality of results of data mining, which may be the difficulty of the evaluation of predicting ability lies[3][4].

As mentioned above, in advance there is no objective and fair data processing standards, which can be used to compare with the results of data mining, therefore, a basis for the judge is bound to be the correct answer by artificially subjective evaluation. Then whether we can use the results of artificially subjective evaluation to be compared with, the answer is usually no. The data mining system with practical applications usually has to deal with numerous data. If we deal with all of the data on artificially statistical analysis, the workload is so huge that testing is often unbearable. Besides, artificially statistical analysis is so subjective that different testers' results are likely to vary from person to person, which will impact on the objectivity and impartiality of the results. Therefore, the comparing test of the results of artificial analysis is very difficult, unless in the face of the small amount of data and having enough human resources to test. But the view that the results of data mining could not be evaluated based on this is also incorrect. If the attention to the testing phase when the degree of how the results of the model match with expectations can be shifted, it will be found that testing to what extent can data mining distinguish from observation points of different types should be the real concern. We can evaluate the predicting ability of data mining model from the perspective of overlapping degree (identification degree) of the data mining results, etc.

2 Overlapping Degree and Its Statistical Test

In the following we will study the structure of overlapping degree of classification data mining algorithms and then propose the statistical tests of overlapping degree[2].

The results of some data mining usually give the probability belonging to a particular type of an observation point so as to achieve the aim of forecast. In order to make the probability more easily to be understood and to study the distribution of observing points with different probability conveniently, we will convert the probability into a continuous numerical, the level of which can show the possibility that observation point falls into any particular category vividly. For example, during the process of the application of data mining in the bank's individual credit risk assessment, customers' default probability is converted to a credit score finally[5]. The higher the score is, the greater the default probability is. In the following, we have introduced the concept of scoring to measure the forecasting ability of models. Because the score is the linear transformation of the prediction probability, the following analysis is applicable to all the assessment of data mining models based on probability.

2.1 Nature of Overlapping Degree

The thinking of overlapping degree is: the difference between distribution of positive point scores and that of negative point scores can be portrayed by the overlapping area of the two parts. First of all, considering the simplest case: both of the point scores are normal distribution, and there is only one intersection point. In this assumption, the distribution density of the two types of points can easily come out.

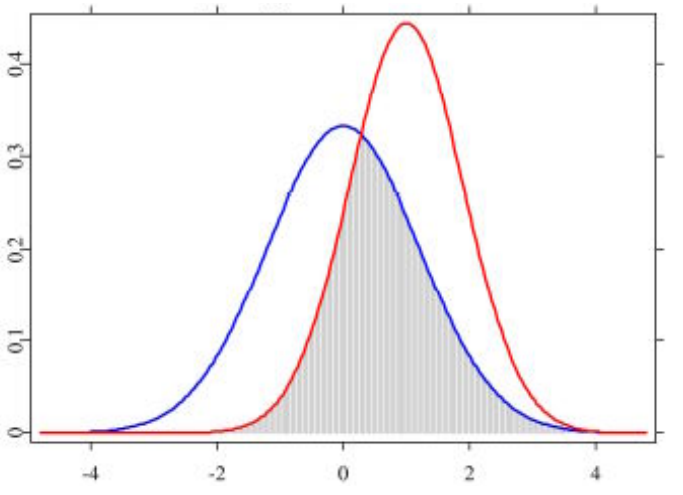


Fig. 1. Diagram of Overlapping Degree

Figure 1 reveals that:

s(s) is used to represent abscissa of the intersection point of the two distributions. S is a critical value for dividing population and when the score of the observation point S>s, the observation point will be predicted as negative. We assume the means of positive point scores and negative points scores are μ_0, μ_1 and the standard deviations are σ_0, σ_1. If it is assumed that $\mu_0 < \mu_1$ and the cumulative distribution functions of S|Y=0 and S|Y=1 are F_0, F_1, the overlapping region O can be calculated by:

$$O = F_1(s) + 1 - F_0(s) \tag{1}$$

When the standard deviations of the two distributions are equal, that is, ($\sigma_0 = \sigma_1$), there exists only one intersection point. In normal distribution, the intersection point coordinate is $s = (\mu_0 + \mu_1) / 2$.

When the standard deviations of the two distributions are not equal, that is, ($\sigma_0 \neq \sigma_1$), there may be one intersection point or two and the abscissa of the intersection point should meet: $f_0(s) = f_1(s)$. Due to normal distribution, the equation has the same solution as formula:

$$\begin{aligned} &s^2(\sigma_1^2 - \sigma_0^2) + 2s(\mu_1\sigma_0^2 - \mu_0\sigma_1^2) + \mu_0^2\sigma_1^2 - \mu_1^2\sigma_0^2 \\ &+2\sigma_1^2\sigma_0^2(\log(\sigma_0) - \sigma_0^2\log(\sigma_1)) = 0 \end{aligned} \tag{2}$$

If we do not assume the scores obey any certain distributions, O can be given in the form of the general non-parameters in the following:

$$o = \int \min\{f_0(s), f_1(s)\} ds \tag{3}$$

In this situation, the existence of more points is allowed. Assuming there is only one point and the relation of score S and the default probability is positive and monotone, the overlapping area is defined as:

$$O_{pos} = \min_s \{F_1(s) + 1 - F_0(s)\} \tag{4}$$

Simultaneously, when the relation of score S and the default probability is negative and monotone, the overlapping area is defined as:

$$O_{neg} = \min_s \{F_0(s) + 1 - F_1(s)\} \tag{5}$$

Therefore, when the relation of score S and the default probability is monotone, there is:

$$O_{neg} = \min\{O_{pos}, O_{neg}\} \tag{6}$$

It can be seen easily that if the two types of points are completely separated, the overlapping area O is 0 and if the two distributions are exactly the same, O is 1, therefore, there is a measure T used to instruct predictive power:

$$T = 1 - O_{mon} = \max_s | F_0(s) - F_1(s) | \tag{7}$$

The value of indicator T is 0~1. When T=1, it means two types of people are completely separated, and when T=0, it stands for the other extreme circumstances. T in the positive and negative correlation cases is as follows:

$$\begin{aligned} T_{pos} &= 1 - O_{pos} = \max_s \{F_0(s) - F_1(s)\} \\ T_{neg} &= 1 - O_{neg} = \max_s \{F_1(s) - F_0(s)\} \end{aligned} \tag{8}$$

In the monotone cases, T can be estimated by the nonparametric estimation (such as empirical distribution function) of the cumulative distribution functions F_0, F_1. In the assumption of normal distribution, O (or T) can be calculated by the parameters of the normal distribution. In a more general assumption of the distribution, O (or T) can be calculated by the nonparametric estimation of probability density of the distribution of scores.

2.2 Statistical Tests of Overlapping Degree

According to the formula 3.8, we have found the indicator T is the same in the form with statistics of Kolmogorov-Smisrnov test, that is, both are counting the greatest vertical distance D(T) of the two probability distributions of the accumulated experience. When the sample size N> 200, statistics D obey normal distribution. Kolmogorov-Smisrnov test is used to detect differences in location, scale, partial or other aspects of the two distributions (any difference between the two distributions), so we can use the statistics to verify whether the data mining model can distinguish observation points of different types. When we choose K-S test to deal with the overlapping degree, we calculate a statistics, that is, testing statistics D based on the sample, and then compare the shape of distribution of samples with normal distribution, so as to arrive at a value p ($0 < p < 1$) , that is, the actual significance level to describe of the degree of suspicion of the idea. If the p value is less than a given significance level (such as 0.05), the original assumption is so suspicious that the data is believed not from the normal distribution, while on the contrary the data is believed from the normal distribution.

We consider such assumption that:

	H_0	H_1	$teststatistics$	$reject\,condition$
(1)	$F_1(x) = F_0(x)$	$F_0(x) > F_1(x)$	$\widehat{T}_{pos} = \max_s\{\widehat{F}_0(s) - \widehat{F}_1(s)\}$	$\widehat{T}_{pos} > \Delta_{n_1,n_0;1-\alpha}$
(2)	$F_1(x) = F_0(x)$	$F_0(x) < F_1(x)$	$\widehat{T}_{neg} = \max_s\{\widehat{F}_1(s) - \widehat{F}_0(s)\}$	$\widehat{T}_{neg} > \Delta_{n_1,n_0;1-\alpha}$
(3)	$F_1(x) = F_0(x)$	$F_1(x) \neq F_0(x)$	$\widehat{T} = \max_s \mid \widehat{F}_0(s) - \widehat{F}_1(s) \mid$	$\widehat{T} > \Delta_{n_1,n_0;1-\alpha/2}$

We use the statistics from (1) (2) to test whether the distributions of F_0, F_1 are the same. The definition of refusal conditions is as follows:

$$\Delta_{n_1,n_0;1-\alpha} = \Delta_{q;1-\alpha} \qquad q = \left|\frac{n_o \cdot n_1}{n_o + n_1}\right|$$

n_0,n_1 stand for the number of positive and negative points and α is a given confidence level. When n_1 drops, the critical value for refusing follows up, but the value of the statistics does not have such a change. This shows that when the statistics is given, the lower the ratio of the negative points is, the more difficult zero assumption is rejected.

3 Lorenz Curve, Gini Coefficient and ROC Curve

Another widely used indicator, which is the measure of the credit score performance, is AR (accuracy ratio) based on Lorenz curve and Gini coefficient. Lorenz curve is also called the choice curve, which depicts the relation between the distribution of all the observation point scores S and that of the negative point scores S (Y = 1) so that we can observe the different distribution points of difference through the image. Figure 2 reveals that Lorenz curve horizontal axis OP and vertical axis OL show respectively all the cumulative percentage of observation and the corresponding cumulative percentage of "negative points" after the sort by points. And diagonal OC is the average diagonal line and the broken line OPC is uneven line [2].

In order to link with the cumulative distribution, we assume a negative score:

$$R= -S$$

Then the coordinate of Lorenz curve can be expressed as:

$$\{L_1(r), L_2(r)\} = \{P(R < r), P(R < r \mid Y = 1)\}, \quad r \in (-\infty, \infty) \tag{9}$$

Because $P(R < r) = 1 - F(s)$, which is equivalent to

$$L(s) = \{L_1(s), L_2(s)\} = \{1 - F(s), 1 - F_1(s)\}, \quad s \in (-\infty, \infty) \tag{10}$$

Lorenz curve can be estimated by the cumulative distribution function of experience. The best Lorenz curve corresponds to the scores that can separate “positive” points from “negative” points completely:

$$L_{opt}(s) = \{1 - F(s), g(1 - F(s))\}, \quad s \in (-\infty, \infty) \tag{11}$$

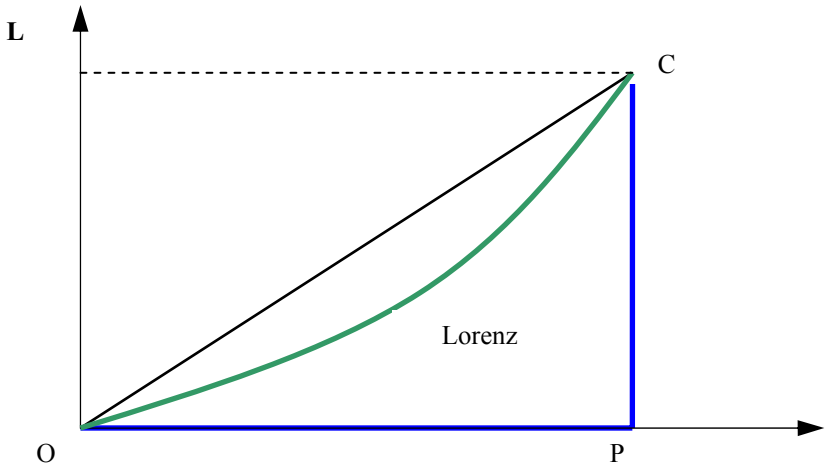


Fig. 2 Diagram of Lorenz Curve

$$g(x) = \begin{cases} \dfrac{x}{P\ (Y\ =\ 1)} & 0 < x \le P(Y\ =\ 1) \\ 1 & P(Y\ =\ 1) < x \le 1 \end{cases} \tag{12}$$

The corresponding scores of the Lorenz curve coinciding with the diagonal completely are completely random, which means scores have nothing to do with the level of customers. So Lorenz curve can compare the performance of categories: good classification score is close to the optimal curve and the scores with weak distinguishing ability is close to the diagonal.

Lorenz curve can depict distinguishing ability of results of data mining intuitively and Gini coefficient can quantify the performance of credit scoring, which shows 2 times of the area of the graphics surrounded by Lorenz curves and diagonal:

$$G = 2\int_{+\infty}^{-\infty} \{1 - F_1(s)\} d\{1 - F(s)\} - 1 = 1 - 2\int_{-\infty}^{+\infty} F_1(s) d(s) \tag{13}$$

For the optimal Lorenz curve is the optimal Gini coefficient, the next formula is given:

$$G_{opt}\ =\ P(Y\ =\ 0) = 1 - P(Y\ =\ 1) \tag{14}$$

In fact accuracy rate (AR) refers to the ratio of Gini coefficient and Gini coefficient when Lorenz curve is the optimal. AR is defined as:

$$A\,R\ =\ \frac{G}{G_{opt}} = \frac{G}{1 - P(Y\ =\ 1)} \tag{15}$$

As we can see from the analysis above, when Lorenz curve is strictly convex, and the relation of credit score S and the default probability Y is monotone and positive (the higher the score is, the greater the default probability is), AR is defined between 0 and 1.Supposing the relation of credit score S and the default probability Y is

monotone and negative, AR may be negative, which means we have to add a symbol to get a positive.

A curve that is similar to the Lorenz curve is ROC curve (receiver operating characteristic curve), the coordinate of ROC curve is defined as:

$$R(s) = \{1 - F_0(s), 1 - F_1(s)\} \tag{16}$$

Compared with the Lorenz curve, the ordinates of ROC curve obey the cumulative distribution of positive points, rather than the cumulative distribution of all the observation points. When the negative points take up a very small proportion, the shape of the two curves are very similar because F roughly equals to F_0, making Lorenz curve and ROC are very similar in shape. As the same with Lorenz curve, the optimal ROC curve corresponds to the scores that can separate the "positive" points from the "negative" ones totally, so the vertex coordinate of optimal ROC curve is (0,0) (1,0) (1,1).

4 AUC Index and Its Nonparametric Test

4.1 AUC Index

In order to quantify the difference between F_0 and F_1, the concept of AUC (area under curve) is introduced[2]:

$$AUC = \int_{+\infty}^{-\infty} \{1 - F_1(s)\} d\{1 - F_0(s)\} = 1 - \int_{-\infty}^{+\infty} F_1(s) dF_0(s) \tag{17}$$

When AUC = 0, there is no difference between F_0 and F_1, and when AUC = 1, the difference between F_0 and F_1 is the maximum. It should be noted that: there is linear correlation between AUC and AR, which is proved by literature [6]:

$$AR = 2AUC - 1 \tag{18}$$

From the definition of Gini coefficient, we can get:

$$\begin{aligned} \frac{1\text{-G}}{2} &= \int_{+\infty}^{-\infty} F_1(s) d(s) = \int_{-\infty}^{+\infty} F_1(s) d\{P(Y=0)F_0(s) + P(Y=1)F_1(s)\} \\ &= P(Y=0)\int_{+\infty}^{-\infty} F_1(s) dF_0(s) + P(Y=1)\int_{+\infty}^{-\infty} F_1(s) dF_1(s) \\ &= P(Y=0)(1 - AUC) + P(Y=1)\cdot\frac{1}{2} = \frac{P(Y=0)}{2} - P(Y=0)\cdot AUC + \frac{1}{2} \end{aligned} \tag{19}$$

So

$$\text{G} = 2AUC \cdot P(Y=0) - P(Y=0). \tag{20}$$

By substitution of G in $AR=G / P(Y=0)$, the result is got.

This shows that the result must be the same whether use AR or AUC to evaluate the scores of different models to be good or bad.

4.2 Nonparametric Test of AUC

In statistical tests, when the overall distribution type is known, the statistical method which uses indexes of samples to make inference or hypothesis testing of the overall parameters is called parametric test; when the overall distribution is unknown, nonparametric test can be used. Many of the classic nonparametric tests such as Wilcoxon rank test and Mann-Withney U test, can verify whether two distributions are from the same collectivity. [7] Two-sample Wilcoxon (or Mann-Whitney) does not have the premise of normality and the only requirement is that the sample is from the same conventionally continuous distribution when the assumption is valid. Wilcoxon test is symmetric test, testing whether the symmetric center of the difference collectivity is 0, and thus infer whether the two samples are from collectivities with the same central location[8][9].

Wilcoxon rank sum test should be applied to compare the information from two samples. The basic idea is: If the test supports the assumption, the rank sum of the two groups should not make that much difference. The basic steps are:

(1) Establish assumptions;
    H0: the same overall distribution of two groups;
    H1: the different locations of overall distribution of two groups;
(2) Rank for the two mixed groups;
(3) Take rank sum of the group whose sample size is the smallest as the test statistics T;
(4) Assuming sample size of the smaller group as n_1, check the critical value sheet using the difference between the two sample size, that is , n_2 - n_1 and T value;
(5) Make conclusions based on P value.

When the sample size is large, normal approximation should be applied to u test. When the same rank is much, correction formula should be applied to u value.

Applying the test methods to evaluate the results of data mining, we get the method to construct u statistics:

$$U = number\ of\{s_{i1} > s_{j0}\} \tag{21}$$

When the positive and negative points are distinguished completely, U= $n_0\ n_1$ is defined, but if they cannot be distinguished at all, in other words, there is no correlation between score S and the default label variable Y, and then the probability of event $S_{i1} > S_{j0}$ is 1/2 $n_0\ n_1$.So $U / n_0 n_1$ statistics can be estimated as follows:

$$\widetilde{U} = P\{(S \mid Y = 1) > (S \mid Y = 0)\} = \int \{1 - F_1(s)\} = AUC \tag{22}$$

So

$$U = \left(\frac{\widehat{AR} + 1}{2}\right) \cdot n_0 \cdot n_1 \tag{23}$$

When the distribution of scores is not continuous, the relation between U and AUC also set up. However, customers with the same scores mustn't be default customers, so $S_{i1} = S_{j0}$ should also be considered by U statistics.

$$P\{(S \mid Y = 0) > (S \mid Y = 1)\} + \frac{1}{2} P\{(S \mid Y = 0) = (S \mid Y = 1)\} \tag{24}$$

In 1975 Lehamann proved that based on the assumption of $F_1(x) = F_0(x)$, U obeys gradual normal distribution when the sample size is large. We can study the following assumptions:

	H_0	H_1	$Test\ Statistic$	$Reject\ condition$
(1)	$F_1(x) = F_0(x)$	$F_0(x) > F_1(x)$	U	$U > k_{n_0,n_1;1-\alpha}$
(2)	$F_1(x) = F_0(x)$	$F_0(x) < F_1(x)$	U	$U < n_0, n_1 - k_{n_0,n_1;1-\alpha}$

The critical value is:

$$k_{n_0,n_1;1-\alpha} = \frac{n_0, n_1}{2} + u_{1-\alpha} \cdot \sqrt{\frac{1}{12} \cdot n_0 \cdot n_1 \cdot (n_0 + n_1 + 1)} \tag{25}$$

Known by the formula above, both the critical value and the statistics U reduce when n_1 drops, so we can not draw the conclusion that zero assumption is more difficult to be rejected when negative points take up a small proportion.

5 Conclusions

In order to test to what extent can data mining distinguish from observation points of different types, the indicators that can measure the difference between the distribution of positive and negative point scores are raised. First of all, we use the overlapping area of two types of point distributions, that is, overlapping degree, to describe the difference, and discuss the nature of overlapping degree. We found that overlapping degree is similar with K-S statistics in the form of measurement, so we use K-S statistics to examine whether the results of data mining model distinguish between "positive" and "negative" points in the given level of confidence .At the same time we have found when the statistics is determined, the smaller the negative points take up the proportion, the more difficult the zero assumption is rejected; Secondly, we put forward the image and quantitative indicators with the ability to distinguish different models: Lorenz curve, Gini coefficient, AR, as well as the similar ROC curve and AUC. We have proved AUC and AR are completely linear related; Finally, we construct the nonparametric statistics of AUC, however, the difference of K-S is that we can not draw the conclusion that zero assumption is more difficult to be rejected when negative points take up a smaller proportion.

Acknowledgements

This research has been partially supported by a grant from National Natural Science Foundation of China (#70501030, #70621001, #90718042) and Beijing Natural Science Foundation (#9073020).

References

1. Padmanabhan, B., Tuzhilin, A.: Knowledge refinement based on the discovery of unexpected patterns in data mining. Decision Support Systems (1999)
2. Wei, J.: Objective Measurements of Intelligent Knowledge (Master Dissertation), Graduate University of Chinesa Academy of Sciences (2008)
3. Sveiby, K.E.: The New Organizational Wealth Managing and Measuring Knowledge-based Assets. Berrett-Koehle~Publishers, San Francisco (1997)
4. Demsar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. Journal of Machine Learning Research 7, 1–30 (2006)
5. Infield, N.: Capitalizing on knowledge. Information World Review (1997)
6. Rasero, B.C.: Statistical Aspects Of Setting Up A Credit Rating System (2003)
7. Shannon, C.E.: The Mathematical Theory of Communication. BSTJ (1948)
8. Fahrmeir, L., Hamerle, A., Tutz, G.: Multivariate Statistische Verfahren. Walter de Gruyter, Berlin (1996) (in German)
9. Michie, et al.: Machine learning, neural and statistical classification. Ells Horwood, Chichester (1994)

Maximum Expected Utility of Markovian Predicted Wealth

Enrico Angelelli[1] and Sergio Ortobelli Lozza[2]

[1] University of Brescia, Department MQ, Contrada Santa Chiara, 50, 25122 Brescia, Italy
[2] University of Bergamo, Department MSIA, Via die Caniana, 2, 24127 Bergamo, Italy

Abstract. This paper proposes an ex-post comparison of portfolio selection strategies based on the assumption that the portfolio returns evolve as Markov processes. Thus we propose the comparison of the ex-post final wealth obtained with the maximization of the expected negative exponential utility and expected power utility for different risk aversion parameters. In particular, we consider strategies where the investors recalibrate their portfolios at a fixed temporal horizon and we compare the wealth obtained either under the assumption that returns follow a Markov chain or under the assumption we have independent identically distributed data. Thus, we implement an heuristic algorithm for the global optimum in order to overcome the intrinsic computational complexity of the proposed Markovian models.

Keywords: Markov chains, expected utility, portfolio strategies, heuristic, computational complexity.

1 Introduction

In this paper, we model the return portfolios with a Markov chain. Under this distributional hypothesis we compare expected utility portfolio strategies with the assumption that returns are independent identically distributed.

The Markovian hypothesis have been widely used in financial modeling. In particular, in option theory, portfolio theory and risk management theory most of the parametric processes used are Markov processes (for portfolio models see, among others, Staino et al. (2007), Rachev et al. (2007), for option pricing models see, among others, Cox et al. (1979), De Giovanni et al. (2008), Iaquinta and Ortobelli (2008), for risk management models see, among others, Longerstaey and Zangari (1996), Lamantia et al. (2006b). In addition, using the methodology proposed by Christoffersen (1998) we can easily show that the Markovian hypothesis of asset returns cannot be rejected (see Lamantia et al. (2006a)). However, even if most of the parametric processes used in financial applications are Markov processes, only recently it has been shown that we can easily maximize inter-temporal performance measures assuming return portfolios following a Markov chain (see Angelelli and Ortobelli (2008)). In this paper we first propose some algorithms that reduce the complexity of the portfolio selection problems based on the Markovianity of the gross

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 588–597, 2009.

returns. In particular, we use the method discussed by Iaquinta and Ortobelli (2006) for non parametric Markovian processes where the transition matrix depends directly on the portfolio weights. This algorithm permits to predict future asset returns and their distributions in polynomial computational times. However, the dependence on the portfolio weights of the transition matrix implies that the computational complexity of these portfolio problems is much higher than assuming that historical observations of returns are independent identically distributed. As a matter of fact, if we use classic methods for global optimum (such as simulated annealing type algorithms see Leccadito *et al.* (2007)) we cannot solve these problems in reasonable computational times. In order to reduce the computational complexity of these portfolio selection strategies, we use the optimization heuristic proposed by Angelelli and Ortobelli (2008). That algorithm permits to check the *n*-dimensional simplex to approximate the global optimum. Secondly, we propose an empirical comparison among portfolio selection strategies based on the optimization of expected utility of future wealth. We use the negative exponential utility and the power utility with different degrees of risk aversion. We propose an ex-post analysis where we compare the sample path of wealth obtained assuming that the investors recalibrate their portfolios at a fixed temporal horizon. Since any of these portfolio strategies is based on the estimation of the distribution of the returns at future times, we get a substantial difference when portfolio selection strategies are developed using the Markovian assumption respect to those based on the assumption that returns are independent identically distributed. So, when we apply Markovian strategies on twenty components of the Dow Jones Industrials, we show that we always get higher returns with respect to returns obtained by means of classic strategies.

The paper is organized as follows. In Section 2 we show how to model non parametric Markov chains and we formalize the maximum expected utility problem with Markov chains. In Section 3 we discuss the ex-post empirical comparison. In the last Section, we briefly summarize the paper.

2 Maximum Expected Utility with Non Parametric Markov Processes

In this section we deal the portfolio selection problem among *n* risky assets with gross returns $z_{t+1} = [z_{1,t+1}, \ldots, z_{n,t+1}]'$ assuming that the portfolio process is described by a homogeneous Markov chain with *N* states. In particular, we assume that investors want to maximize their utility of wealth at a given future date T. We denote by $x = [x_1, \ldots, x_n]'$ the vector of the positions taken in the *n* risky assets, then the portfolio return during the period [*t*, *t*+1] is given by $z_{(x),t+1} = x' z_{t+1} = \sum_{i=1}^{n} x_i z_{i,t+1}$.

2.1 The Markovian Evolution Process

Next, we consider the range $(\min_k z_{(x),k}; \max_k z_{(x),k})$ of the portfolio gross returns, where $z_{(x),k}$ is the *k*-th past observation of the portfolio $z_{(x)}$. Without loss of generality

we assume that the N states $z_{(x)}^{(i)}$ of portfolio gross return are ordered as follows $z_{(x)}^{(i)} > z_{(x)}^{(i+1)}$ for $i = 1,...,N-1$. Since we want to have a recombining tree of the Markov chain, we first divide the portfolio support $(\min_k z_{(x),k}; \max_k z_{(x),k})$ in N intervals $(a_{(x),i}; a_{(x),i-1})$ where $a_{(x),i} = \left(\frac{\min_k z_{(x),k}}{\max_k z_{(x),k}}\right)^{i/N} \cdot \max_k z_{(x),k}$, $i = 0,1,\ldots,N$ is decreasing with index i. Then, we compute the return associated to each state as the geometric average of the extremes of the interval $(a_{(x),i}; a_{(x),i-1})$, that is

$$z_{(x)}^{(i)} := \sqrt{a_{(x),i} a_{(x),i-1}} = \max_k z_{(x),k} \left(\frac{\max z_{(x),k}}{\min z_{(x),k}}\right)^{\left(\frac{1-2i}{2N}\right)}, i = 1,2,...,N. \quad (1)$$

Consequently $z_{(x)}^{(i)} = z_{(x)}^{(1)} u^{1-i}$, where $u = \left(\frac{\max z_{(x),k}}{\min z_{(x),k}}\right)^{1/N} > 1$. Let us assume that the initial wealth W_0 at time 0 is equal to 1, while for each possible wealth W_t at time t we have N possible different values $W_{t+1} = W_t z_{(x)}^{(i)}$ (i=1,…,N) at time t+1. Thanks to the recombining effect of the Markov chain we have $1 + k(N-1)$ possible values after k steps of wealth $W_k(x)$ that are given by the formula $w_{(x)}^{(i,k)} = (z_{(x)}^{(1)})^k u^{(1-i)}$ $i = 1,...,(N-1)k+1$, where the i-th node at time k of the Markovian tree corresponds to wealth $w_{(x)}^{(i,k)}$. Moreover, all possible values of the random wealth $W_k(x)$ can be stored in a matrix with k columns and $1 + k(N-1)$ rows resulting in $O(Nk^2)$ memory space requirement. Since we assume homogeneous Markov chain the transition matrix $P = [p_{i,j}]$ does not depend on time and the entries $p_{i,j}$ are estimated using the maximum likelihood estimates $\hat{p}_{i,j} = \frac{\pi_{ij}(K)}{\pi_i(K)}$, where $\pi_{ij}(K)$ is the number of observations (out of K observations) that transit from the i-th state to the j-th state and $\pi_i(K)$ is the number of observations (out of K observations) in the i-th state (see D'Amico (2003) for the statistical properties of these estimators). Following the idea of Iaquinta and Ortobelli (2006) we can compute the distribution function of the future gross returns. In particular, as shown by Angelelli and Ortobelli (2008), the $(N-1)k+1$ dimensional vector $p^{(k)}$ (representing the unconditional distribution at a given time $k = 0,1,2,...,T$ of wealth $W_k(x)$) can be computed by means of a sequence of matrixes $\left\{Q^{(k)}\right\}_{k=0,1,...,T}$, where $Q^{(k)} = [q_{i,j}^{(k)}]_{\substack{1 \le i \le (N-1)k+1 \\ 1 \le j \le N}}$ and $q_{i,j}^{(k)}$ is the unconditional probability at time k to obtain the wealth $w_{(x)}^{(i,k)}$ and to be in the state $z_{(x)}^{(j)}$. In particular, $Q^{(0)} = [p_1,...,p_N]$, where p_i is the unconditional probability to be in the i-th state at time *0*. Thus, $p^{(0)} = 1 = Q^{(0)} \cdot \mathbf{1}_N$, where $\mathbf{1}_N$ is the unity vector column. In general, for $k = 1,...,T$, the vector $p^{(k)}$ is given by $p^{(k)} = Q^{(k)} \cdot \mathbf{1}_N$, where $Q^{(k)}$ is

recursively defined as $Q^{(k)} = \texttt{diagM}(Q^{(k-1)} \cdot P)$ being **diagM** a linear operator defined for any $m,n \in N$ as **diagM:** $R^{mn} \to R^{(m+n-1)n}$ that at any $m \times n$ matrix $A = [a_{ij}]$ associates the $(m+n-1)\times n$ matrix obtained by simply shifting down the *j*-th column by $(j-1)$ rows (see Iaquinta and Ortobelli (2006), Angelelli and Ortobelli (2008) for further details). The matrix $Q^{(k)}$ is the so called *unconditional evolution matrix* of the Markov chain or simply *evolution matrix.* Moreover, the algorithm to compute the probabilities has a computational complexity of $O(N^3k^2)$.

2.2 The Portfolio Selection Problem

The static portfolio selection problem when no short sales are allowed, can be represented as the maximization of the expected utility applied to the random portfolio of gross returns $z_{(x),t+1}$ subject to the portfolio weights belonging to the *n*-dimensional simplex $S = \left\{ x \in \mathrm{R}^n \mid \sum_{i=1}^n x_i = 1; x_i \geq 0 \right\}$, i.e., $\max_{x \in S} E(u(z_{(x),t+1}))$, for a given utility function *u.* This represents the classic *myopic utility functional* that does not use the time evolution of the wealth process. In a dynamic context we consider an initial wealth $W_0 = 1$ and all admissible wealth processes $W(x) = \{W_t(x)\}_{t \geq 0}$ depending by an initial portfolio $x \in S$ are defined on a filtered probability space $\left(\Omega, \Im, (\Im_t)_{0 \leq t \leq \infty}, P\right)$. In this case we can distinguish two cases: the case where the investors recalibrate the portfolio at some given date *T* (*European portfolio selection strategies*) and the case where the investors recalibrate the portfolio at some given date $t \leq T$ if some particular events $A_t \in \Im_t$ happen (*American portfolio selection strategies*). In this paper we deal only European portfolio selection strategies where investors recalibrate their portfolio every *T* periods solving the problem:

$$\max_{x \in S} E\left(u(W_T(x))\right). \tag{2}$$

According to Angelelli and Ortobelli (2008) definition we call *OA expected utility* the above functional $E(u(W_T(x)))$ when it is computed under the assumption that the gross return of each portfolio follows a Markov chain with *N* states. The European OA expected utility is given by

$$E(u(W_T(x))) = u\left(\hat{W}_T(x)\right) \cdot Q^{(T)} \cdot \mathbf{1}_N = u\left(\hat{W}(x)\right) \cdot p^{(T)}, \tag{3}$$

where $\hat{W}_T(x) = \left[w_{(x)}^{(1,T)}, ..., w_{(x)}^{(N-1)T+1,T)} \right]$ is the $(N-1)T+1$ dimensional vector of the final wealth and $u\left(\hat{W}_T(x)\right) = \left[u(w_{(x)}^{(1,T)}), ..., u(w_{(x)}^{(N-1)T+1,T)}) \right]$. Since Angelelli and Ortobelli (2008) have shown that standard optimization algorithms are not adequately suited to solve the global optimization problem (2) of OA expected utility, we use the same optimization heuristic proposed by Angelelli and Ortobelli (2008) to solve portfolio optimization problems. So, starting by an initial feasible portfolio solution $\bar{x}$, the heuristic algorithm tries to iteratively update the current solution by a better one. Improving

solutions, if any, are searched on a predefined grid of points fixed on the directions $x - e_i$ for $i = 1, 2, ..., n$, where x is the current portfolio and e_i is the portfolio where the share of asset i is equal to 1 and all other assets have share equal to 0. If a better solution is found on a search direction the current solution is updated and the search is continued from the new one. If no direction provides an improved solution the search ends. Next, we recall some empirical results provided by Angelelli and Ortobelli (2008), who tested the performance of the optimization heuristic algorithm versus function **fmincon** provided with the optimization toolbox of MATLAB. The results are synthesized in Table 1 that reports the percentage in average of variations :

- of the estimated function $\Delta f = \frac{f_{\text{heuristic}} - f_{\text{fmincon}}}{f_{\text{fmincon}}}$, where $f_{[]}$ represents the optimal objective function obtained using the [ ] algorithm ([ ] can be either fmincon or the heuristic);
- of the time $\Delta t = \frac{Timef_{\text{heuristic}} - Timef_{\text{fmincon}}}{Timef_{\text{fmincon}}}$ where $Timef_{[]}$ represents the computational time necessary to optimize the objective function using the [ ] algorithm;
- of the portfolio weights $\Delta x = \sum_{i=1}^{n} \left| x^{(i)}_{\text{heuristic}} - x^{(i)}_{\text{fmincon}} \right|$ where $x^{(i)}_{[]}$ represents the i-th optimal weight obtained using the [ ] algorithm.

Table 1. Performance comparison between fmincon and the optimization heuristic (see Angelelli and Ortobelli (2008) for a definition of these strategies)

Functional	Δf	Δt	Δx
Myopic Sharpe	-0.002%	502.963%	0.010
OA-Sharpe	163.084%	328.202%	1.447
Myopic Rachev	2.696%	213.160%	0.774
OA-Rachev	15465.330%	240.005%	1.681

Table 1 underlines the limit of **fmincon Matlab** procedure to approximate a global optimum when the functionals admit many local maxima. These results tell us that the heuristic algorithm generally needs more computational time of **fmincon Matlab** procedure. However, we have a significant improvement in terms of objective function and portfolio weights when we use the heuristic. Moreover, the heuristic well approximates the optimum when this is unique; indeed there is just a little difference with the myopic Sharpe functional in terms of values and portfolios. From the results we deduce that the **fmincon** procedure can be used only for myopic strategies that admit an unique optimum (such as the myopic Sharpe strategy). Thus, as suggested by Angelelli and Ortobelli (2008), the main advantages of this algorithm are:

1) The algorithm permits to approximate the global optimum with a given error when the objective function is a non-constant concave function (the optimum is unique) and some particular lines are not contour lines of the objective function.
2) The algorithm permits to explore the whole simplex.
3) The computational complexity is much less than that of classic algorithms for global optimum such as Simulated Annealing type algorithms.

3 An Ex-post Comparison among OA Portfolio Strategies Based on the Maximum Expected Utility

In this section, we propose an ex post comparison among European OA expected utility strategies and the myopic ones. In the empirical comparisons, we consider the optimal allocation among 20 assets components of the Dow Jones Industrials[1] on the period from 1/3/1985 till 5/1/2008 for a total of 5884 daily observations. The work of Kondor et al. (2007) on the sensitivity to estimation error of portfolios optimized under various risk measures suggests that we need a large number of observations when we want to propose portfolio models considering rare events. As a matter of fact, Papp et al. (2005), Kondor et al. (2007) have shown that we could loose robustness of the approximations if the number of observations is not adequate to the number of assets. In addition, some empirical experiments show that, if we increase the number of the states, we need an increasing number of observations. For this reason we forecasted the future wealth using a non parametric Markov chain with only few states N=3, N=5 states and $K = 2000$ historical observations. We assume investors recalibrate the portfolio every $T = 60$ days starting from 1/3/1985. The comparison consists in the ex post evaluation of the wealth produced by the strategies. We compare the performance of myopic and OA expected utility strategies based on the following HARA utility functions:

1) negative exponential utility function:

$$u(W) = -\exp(-aW)\text{ ; with } \quad a\text{=1, 5, 10, 15, 20.}$$

2) power utility function:

$$u(W) = \frac{W^g}{g}\text{; with } \quad g\text{=-1, -0.6, -0.2, 0.2, 0.6, 1, 1.4, 1.8, 2.2, 2.6, 3.}$$

With myopic strategies the expected utility of each portfolio is approximated considering the last $K = 2000$ observations and computing $E(u(x'z_{t+1})) \approx \frac{1}{K}\sum_{t=1}^{K} u\left(x'z_{t+1}\right)$. For each strategy, we consider an initial wealth $W_0 = 1$ at the date 1/3/1985, and at the k-th recalibration ($k = 0,1,2,...$), the investor should solve:

$$\begin{aligned} &\max_{x^{(k)}} E(u(\hat{W}_{t_k+60}(x^{(k)}))) \\ &\text{s.t.} \\ &\left(x^{(k)}\right)' e = 1, \\ &x_i^{(k)} \geq 0; \quad i = 1,\ldots,n, \end{aligned} \tag{4}$$

[1] We used the following components: 3M Company, Alcoa Inc, American Express, AT&T, Boeing Co, Caterpillar Inc, Coca Cola, Du Pont, Exxon Mobil, General Electric, General Motors, Hewlett Packard, IBM, Johnson and Johnson, McDonalds, Merck, Procter Gamble, United technologies, Wal Mart Stores, Walt Disney.

where $\hat{W}_{t_k+60}$ is the forecasted wealth at time t_{k+1}. So, the ex-post final wealth is given by $W_{t_{k+1}} = W_{t_k}\left(\left(x_M^{(k)}\right)' z^{(ex\ post)}\right)$, where $z^{(ex\ post)}$ is the vector of observed gross returns between t_k and $t_{k+1} = t_k + 60$.

Table 2. Final wealth obtained at date 5/1/2008 using myopic and Markovian strategies and maximizing the expected power utility

Parameter	HARA Power Utility	OA-HARA-power utility Markovian strategies	
g	Myopic strategy	states=3	states=5
-1	3.8135	10.8913	9.1947
-0.6	3.9096	11.102	9.466
-0.2	3.973	11.2265	9.5626
0.2	3.5307	11.3135	9.104
0.6	2.0762	11.2862	9.142
1	1.5423	11.3808	7.9091
1.4	2.2367	11.5531	7.7919
1.8	2.0602	11.324	7.7235
2.2	1.9872	11.2224	8.131
2.6	6.0933	10.9617	7.623
3	3.4551	11.1149	7.9061

Table 3. Final wealth obtained at date 5/1/2008 using myopic and Markovian strategies and maximizing the expected negative exponential utility

	HARA negative exponential utility	OA-HARA negative exponential utility Markovian strategies	
a	Myopic strategies	states=3	states=5
1	3.927	11.1103	9.6179
5	4.187	12.2377	9.6295
10	5.087	10.9565	10.9592
15	5.494	8.8739	12.414
20	5.950	9.4837	7.7379

The output of this analysis is represented in Tables 2, 3, Fig. 1, and Fig. 2. Tables 2 and 3 show the ex-post final wealth at date 5/1/2008 obtained with myopic and Markovian strategies and maximizing expected power utility and expected negative exponential utility. We observe that always the Markovian strategies perform better than myopic strategies. Moreover we also observe that we get better results using three states. We believe that this fact can be justified by a more robust approximation of the forecasted final wealth (see Papp et al. (2005), Kondor et al. (2007)). These results are further confirmed by Fig. 1, and Fig. 2 that describe the ex post sample paths of final

wealth. Figures 1 and 2 show better performance of OA expected utility strategies respectively for power utility with risk aversion parameter g= -0.2 and with negative exponential utility with parameter a= 10.

This empirical comparison suggests the use of OA type strategies since with these strategies we get in some cases even three times the final wealth we get with the analogous myopic strategies.

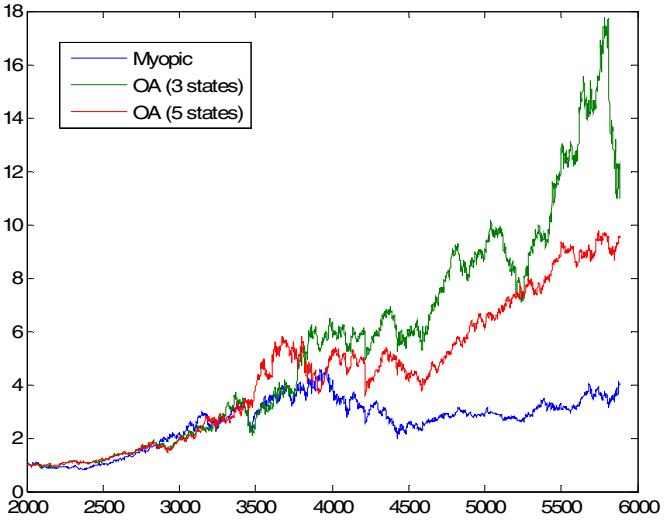


Fig. 1. Performances obtained with HARA power utility (g= -0.2)

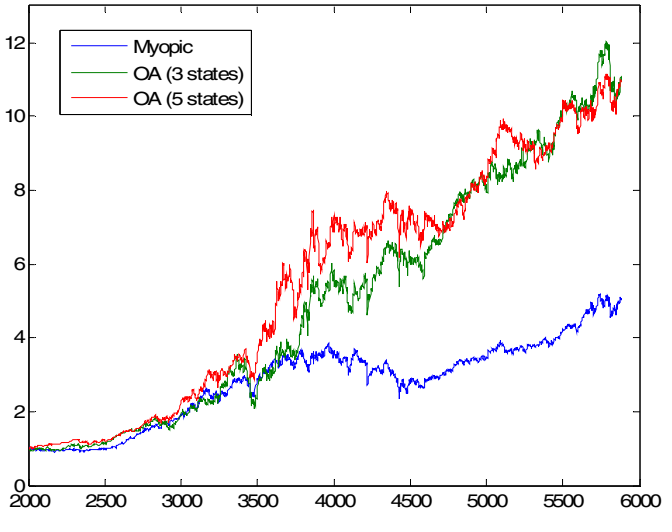


Fig. 2. Performances obtained with HARA negative exponential utility (a=10)

4 Concluding Remarks

This paper analyzes the impact of Markovianity in optimal portfolio choices. We examine how to approximate non parametric Markov processes and we deal the computational complexity of these portfolio selection problems. Thus we propose algorithms that permit to solve computationally complex problems in acceptable computational times. Secondly, we propose an empirical comparison among the myopic portfolio selection models and the Markovian ones. The ex-post empirical comparison among classic approaches and those based on Markovian trees shows the greater predictable capacity of the latter.

The contribution of this paper consists in the computational accessible methodology to solve dynamic expected utility portfolio problems.

Acknowledgments

The authors thank for grants COFIN 60% 2008 and seminar audiences at AMASES 2008 (Trento, Italy), and at X Workshop in Quantitative Finance 2009 (Milan, Italy).

References

1. Angelelli, E., Ortobelli, S.: American and European Portfolio Selection Strategies: The Markovian Approach. In: Columbus, F. (ed.) Financial Hedging: Risks, Strategies and Performance, ch. 5. Nova Science Publishers, New York (2009) (forthcoming)
2. Cox, J.C., Ross, S.A., Rubinstein, M.: Option Pricing: a Simplified Approach. Journal of Financial Economics 7, 229–263 (1979)
3. Christoffersen, P.: Evaluating Interval Forecasts. International Economic Review 39(4), 841–862 (1998)
4. D'Amico, G.: Markov Chain European Option: Statistical Estimation. Technical Report, Università di Roma "Sapienza" (2003)
5. De Giovanni, D., Ortobelli, S., Rachev, S.T.: Delta Hedging Strategies Comparison. European Journal of Operational Research 185(3), 1615–1631 (2008)
6. Kondor, I., Pafka, S., Nagy, G.: Noise Sensitivity of Portfolio Selection under Various Risk Measures. Journal of Banking and Finance 31, 1545–1573 (2007)
7. Iaquinta, G., Ortobelli, S.: Distributional Approximation of Asset Returns with Non Parametric Markovian Trees. International Journal of Computer Science & Network Security 6(11), 69–74 (2006)
8. Iaquinta, G., Ortobelli, S.: Markov Chain Applications to Non Parametric Option Pricing Theory. International Journal of Computer Science & Network Security 8(6), 199–208 (2008)
9. Lamantia, F., Ortobelli, S., Rachev, S.T.: An Empirical Comparison among VaR Models and Time Rules with Elliptical and Stable Distributed Returns. Investment Management and Financial Innovations 3, 8–29 (2006a)
10. Lamantia, F., Ortobelli, S., Rachev, S.T.: VaR, CVaR and Time Rules with Elliptical and Asymmetric Stable Distributed Returns. Investment Management and Financial Innovations 3(4), 19–39 (2006b)

11. Leccadito, A., Ortobelli, S., Russo, E.: Portfolio Selection, VaR and CVaR Models with Markov Chains. International Journal of Computer Science & Network Security 7(6), 115–123 (2007)
12. Longerstaey, J., Zangari, P.: RiskMetrics - Technical Document, 4th edn. J.P. Morgan, New York (1996)
13. Papp, G., Pafka, S., Nowak, M.A., Kondor, I.: Random Matrix Filtering in Portfolio Optimization. ACTA Physica Polonica B 36, 2757–2765 (2005)
14. Rachev, S., Stoyanov, S., Fabozzi, F.: Advanced Stochastic Models Risk Assessment, and Portfolio Optimization: the Ideal Risk, Uncertainty and Performance Measures. John Wiley and Sons, Hoboken (2007)
15. Staino, A., Ortobelli, S., Massabò, I.: A Comparison among Portfolio Selection Models with Subordinated Lévy Processes. International Journal of Computer Science & Network Security 7(7), 224–233 (2007)

Continuous Time Markov Chain Model of Asset Prices Distribution

Eimutis Valakevičius

Kaunas University of Technology, Faculty of Fundamental Sciences,
Studentų st. 50, LT - 51368 Kaunas, Lithuania
eimval@ktu.lt

Abstract. The aim of this paper is to introduce a new model of a financial asset prices distribution. It is known that the probability distribution of an asset prices or returns is unknown in reality. The general model of asset prices based on continuous time Markov chains is proposed. For this reason the interarrivals between two price states are approximated by mixture of exponential distributions. Numerical-analytic approach is used to obtain the probability distribution of asset prices. The developed software allows creating the space of an asset prices, the matrix of transition rates among states, a system of equations to find the steady state probabilities of price states and solves the system of equations by method of imbedded Markov chains.

Keywords: Asset prices distribution, a mixture of exponential distributions, continuous time Markov chain, and numerical-analytic model.

1 Introduction

It should be noted that, in practice, we do not observe asset prices following continuous-variable, continuous-time processes. For example, stock prices are restricted to discrete values and changes can be observed only when exchange is open. Nevertheless, the discrete-variable and continuous-time process proves to be useful model for stock prices.

To price and hedge derivative securities, it is crucial to have a good model of the probability distribution of the underlying product. The most famous continuous time model is the celebrated Black - Scholes model [1] and discrete one is classical Cox-Rubinstein model [2], which uses the normal distribution to fit the log returns of the underlying asset. One of the main problems with these models is that the data suggest that the log returns of stocks are not normally distributed. So other more flexible distributions are needed. Some authors suggest the underlying normal distribution to replace by a more sophisticated one. Examples of such, which can take into account skewness, excess kurtosis and other features, are the Variance Gamma [3], the Normal Inverse Gaussian [4], the CGMY (named after Carr, Geman, Madan and Yor) [5], the Hyperbolic Model [6] and the Meixner [7] distribution. Including such features makes analytic modelling less tractable, and potentially makes numerical modelling a more attractive alternative. In the following sections the algorithm of modeling asset

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 598–605, 2009.

prices by Markov chains is proposed. Armed with the model of price dynamics, an investor can:

- Calculate theoretical prices for derivative securities
- Measure the amount of risk associated with holding risky securities.

2 Markov Property of Stock Prices

The dynamics of asset prices are reflected by uncertain movements of their values over time. Some authors [8, 9] state that Efficient Market Hypothesis (EMH) is one possible reason for the random behaviour of the asset price. The EMH basically states that past history is fully reflected in present prices and markets respond immediately to new information about the asset.

A Markov process is a particular type of stochastic process where only the present value of a variable is relevant for predicting the future. The past history of the variables and the way that that the present has emerged from the past are irrelevant.

Stock prices are usually assumed to follow a Markov process. These processes are important models of security prices, because they are often realistic representation of true prices and yet the Markov property leads to simplified computations. If the stock price follows a Markov process, our predictions of the future should be unaffected by the price one week ago, one month ago, or one year ago. The only relevant piece of information is the price now. Predictions are uncertain and must be expressed in terms of probability distributions. The Markov property implies that the probability distribution of the price at any particular future time is not dependent on the particular path followed by the price in the past.

If stock price process $S = \{S_t, 0 \le t \le T\}$ is Markovian and if we denote by $F = \{F_t, 0 \le t \le T\}$ the natural filtration of S (intuitively, F_t contains all market information up to time t), then we can write for a well-behaved function f: $E[f(S_T) | F_t] = E[f(S_T) | S_t]$. The stock price process takes values in some countable set E, called the state space. If $S_t = j \in E$, we shall say "the process is in state j at time t". The most common situation is for the state to be a scalar, but frequently it is more convenient for the state to be a vector.

3 Approximation of the Probability Distribution of Stock Prices

Our aim is to construct the stock price dynamics as a continuous time Markov chain with countable space of states. To find the space of states and transition rates between them we have to construct price movement distributions up and down for a given stock. To get Markov process the distribution of time length of stock price rising or decreasing must be exponential with parameter μ. Unfortunately, usually it is insufficient, and then a convenient representation for more general distributions is the Coxian formulation [8]. This formulation, by means of fictitious phases, allows the duration of generating stock price rate of transition up or down to be described by a linear combination of stochastic variables. Thus, generation of price movement is a continuous succession of k phases,

each having exponential service time distribution of rate μ_j, $j = 1,2,\cdots,k$. After phase j, a stock price leaves the phases with probability $(1-p_j)$. The stock price can occupy only one phase at a time. Therefore, there can be at most one stock price within the set of phases at any time.

Let us consider a general probability distribution function $G(t)$ of stock prices. Useful approximation of this function can be obtained by the mixture and convolutions of exponential (phase-type) distributions. Then a Markov chain with a countable space of states and continuous time can represent the evolution of stock price dynamics. Suppose we let $m_k, k = \overline{1,3}$ denotes the kth non-central moment, i.e. $E[X^k]$, where X is a random variable of price movement time. Construct a random variable X, which can be represented as:

$$X = \begin{cases} X_1 \text{ with prob. } p_2; \\ X_1 + X_2 \text{ with prob. } p_1 p_2; \\ \qquad\qquad \cdots \\ X_1 + X_2 + \ldots + X_2 \text{ with prob. } p_1^{n-1} p_2; \\ \qquad\qquad \cdots \end{cases}$$

where X_i, $i = 1,2,$ are independent random variables having exponential distributions with means $1/\mu_1$ and $1/\mu_2$ respectively; $p_1 + p_2 = 1$. The random variable X equals to the sum of independent variables with random number N of summands. N is non-negative, integer-valued random variable with $E(N) < \infty$ having geometrical distribution. Its probability density is the following

$$f(x) = p_2\mu_1\left(\frac{\mu_2-\mu_1}{\mu_2 p_2-\mu_1}e^{-\mu_1 x} - \frac{\mu_2 p_1}{\mu_2 p_2-\mu_1}e^{-p_2\mu_2 x}\right)$$

Moment matching is a common method for approximating distributions. Though two-moment approximations are common, they may lead to serious error when the coefficient of variation ν, (the standard deviation divided by the mean) is high. The first three moments of any non degenerate distribution with support on [0,∞) can be matched by the distribution (2).

To obtain the values of the parameters μ_1, μ_2, p_1 and p_2 of approximation, a complex system of non-linear equations needs to be solved:

$$\begin{cases} \dfrac{1!\cdot p_2\mu_1}{\mu_2 p_2-\mu_1}\left(\dfrac{\mu_2-\mu_1}{\mu_1^2} - \dfrac{\mu_2 p_1}{\mu_2^2 p_2^2}\right) = m_1; \\ \dfrac{2!\cdot p_2\mu_1}{\mu_2 p_2-\mu_1}\left(\dfrac{\mu_2-\mu_1}{\mu_1^3} - \dfrac{\mu_2 p_1}{\mu_2^3 p_2^3}\right) = m_2; \\ \dfrac{3!\cdot p_2\mu_1}{\mu_2 p_2-\mu_1}\left(\dfrac{\mu_2-\mu_1}{\mu_1^4} - \dfrac{\mu_2 p_1}{\mu_2^4 p_2^4}\right) = m_3; \\ p_1 + p_2 = 1. \end{cases}$$

The solution of the system is the following:

$$\mu_2 = \frac{g_2 - g_1^2}{g_1^3 - 2g_1 g_2 + g_3}, g_k = \frac{m_k}{k!}, k = \overline{1,3};$$

$$\mu_1 = \frac{1 + \mu_2 g_1 \pm \sqrt{(1-\mu_2 g_1)^2 + 4\mu_2{}^2 \left(g_2 - g_1^2\right)}}{2g_1 - 2\mu_2 \left(g_2 - g_1^2\right)};$$

$$p_1 = \frac{\mu_2(\mu_1 g_1 - 1)}{\mu_2(\mu_1 g_1 - 1) + \mu_1}; \quad p_2 = \frac{\mu_1}{\mu_2(\mu_1 g_1 - 1) + \mu_1}.$$

The exponential stages are shown graphically in Fig.1.

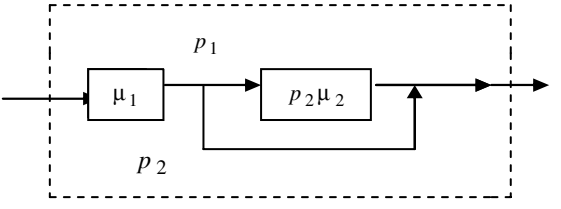


Fig 1. The diagram of two exponential phases

4 Calculation of Steady State Distribution of Markov Chain

In constructing the approximating Markov chain, three decisions need to be made. First, one must choose the set of discrete prices, i.e. $S = \{S_0, S_1, ..., S_n\}$. The second decision is generating the transition rates among states, and the third - calculating steady state distribution of asset prices.

In this section we will use so called the event language to generate the space of stock prices and transition matrix between them. Denote the time elapsed between two consecutive time observations by $\Delta Y = Y_i - Y_{i-1}, i = 1,2,...$

We approximate the distributions of price movement time up and down from the historical data of selected stock prices by the formulas respectively

$$f_u(y) = p_1^u \mu_1^u \left(\frac{\mu_2^u - \mu_1^u}{\mu_2^u p_2^u - \mu_1^u} e^{-\mu_1^u y} - \frac{\mu_2^u p_2^u}{\mu_2^u p_2^u - \mu_1^u} e^{-\mu_2^u p_2^u y} \right)$$

$$f_d(y) = p_1^d \mu_1^d \left(\frac{\mu_2^d - \mu_1^d}{\mu_2^d p_2^d - \mu_1^d} e^{-\mu_1^d y} - \frac{\mu_2^d p_2^d}{\mu_2^d p_2^d - \mu_1^d} e^{-\mu_2^d p_2^d y} \right)$$

A Markov chain with the countable space of states and continuous time can describe the dynamics of stock price movement. To construct a numerical model of the system, the approach proposed in will be applied.

The set of events in the system:

$$E = \{e_1^u, e_2^u, e_3^u, e_4^u, e_1^d, e_2^d, e_3^d, e_4^d\}$$

where

$e_1^{u(d)}$ – beginning of price movement up (down);

$e_2^{u(d)}$ – completed the stage of price movement up (down) with probability $p_1^{u(d)}$ in the first phase;

$e_3^{u(d)}$ – completed the stage of price movement up (down) with probability $p_2^{u(d)}$ in the first phase;

$e_4^{u(d)}$ – completed the stage of price movement up (down) in the second phase;

The set of transition rates:

$Intens = \{\mu_1^u, p_2^u \mu_2^u, \mu_1^d, p_2^d \mu_2^d\}$,

where

$\mu_1^{u(d)}$- rate of price movement up (down) in the first phase;

$\mu_1^{u(d)} p_2^{u(d)}$- rate of price movement up (down) in the second phase;

Let us consider an asset observed on a discrete time scale $\{0,1,\cdots,t,\cdots,T\}, T<\infty$ having S_t as market stock value at time *t*. To model the stochastic process $(S_t, t=0,1,\cdots,T)$ we suppose that the asset has known minimal and maximal values so that the set of all possible values is the closed interval $[S_{\min}, S_{\max}]$. For example, if S_0 is the value of the asset at time 0, we can put

$$S_0 = (S_{\max} + S_{\min})/2$$

$$S_k = S_0 + k\Delta,\ k = 1,\cdots,N$$

$$S_{-k} = S_0 - k\Delta,\ k = 1,\cdots.N$$

$$\Delta = (S_{\max} - S_{\min})/2N$$

N being chosen arbitrarily. This implies the total number of states is $2n + 1$. In what follows, we order these states in the naturally increasing order and use the following notation for the state space:

$$I = \{-N,-(N-1),\cdots,0,1,\cdots,N\}.$$

We can also introduce different step lengths following movements up and down and so consider respectively Δ, Δ'. It is also possible to let $S_{\max} \to \infty$ and $T \to \infty$ particularly to get good approximation results.

To model the dynamics of stock prices we need know the phase in which is the stock price. The state space of the system is completely specified by the set of triples

$$B = \{(b_1, b_2, b_3)\},\ b_1 \in I,$$

where

$$b_2 = \begin{cases} 0, \text{if the stock price is not changing up;} \\ 1, \text{if the stock price is moving up in the first phase;} \\ 2, \text{if the stock price is moving up in the second phase.} \end{cases}$$

$$b_3 = \begin{cases} 0, \text{if the stock price is not changing down;} \\ 3, \text{if the stock price is moving down in the first phase;} \\ 4, \text{if the stock price is moving down in the second phase.} \end{cases}$$

The dynamics of stock price movement can be described in the event language. As an example, the description of the fourth event using pseudo code is represented bellow.

```
e_4^u :      if  b_2 = 2  and  b_1 < n
               then  b_1 ← b_1 + 1; b_2 ← 0
               Intense ← μ_1^u p_1
               end  then
          end if
end  e_2^u
```

The software for automatic construction of numerical models is created [9]. The software consists of:

- The language of a model specification
- A program for automatic generation of all possible states (the set of stock prices) and transition rates among them
- A program for calculation of steady state probabilities of continuous Markov chain.

The states of Markov process are generated from an initial state. All possible transitions from this state are considered. When this step is completed, the current state is marked, and one of the newly obtained states becomes the current state. The generation process terminates when all the states in the list have been marked and no new state is obtained. Let $\Lambda_N = \left(\lambda_{ij}^{(N)}\right)_{N\times N}$ be the matrix of transition rates for a Markov chain on the set of states $S = \{S_0, \ldots, S_N\}$. Then

$$\lambda_{ij}^{(k)} = \lambda_{ij}^{(k+1)} + \frac{\lambda_{i,k+1}^{(k+1)} \lambda_{k+1,j}^{(k+1)}}{\overline{S}_{k+1}^{(k+1)}}, \quad i, j = \overline{0,k}; \quad k = \overline{N-1,1};$$

$$\overline{S}_{k+1}^{(k+1)} = \sum_{\substack{j=0 \\ j\neq k+1}}^{N} \lambda_{k+1,j}^{(k+1)};$$

$$r_1^{(1)} = 1; \quad r_i^{(k+1)} = \begin{cases} r_i^{(k)}, \quad i = \overline{0,k}; \\ \dfrac{\sum_{j=1}^{N-1} r_j^{(k)} \lambda_{ji}^{(k+1)}}{\overline{S}_i^{(k+1)}}, i = k+1, k = \overline{1, N-1}; \end{cases}$$

$$q_i = \frac{r_i^{(N)}}{\sum_{i=1}^{N} r_j^{(N)}}, \quad i = \overline{0,N}.$$

The probabilities of stock price states are calculated by the following formula:

$$p(k) = \sum_{b_2, b_3} q(k, b_2, b_3), k = -N, \cdots, N,$$

where $q(k, b_2, b_3)$ is the probability of the price at the fixed phase.

5 Numerical Example

This section discusses the actual implementation of the model's methodology discussed in the previous sections.

Data used in the analyses is the *Microsoft Corporation* daily log-returns which start from February 2007 and end in May 2007. The statistical analysis showed that the skewness is negative and the kurtosis is higher than three, it can be said that the *Microsoft Corporation* log-returns in the sample period are not drawn the unique normal distribution. Thus it would be challenging to apply numerical-analytic method.

From the observed data we can assume that $S_{\max} = 28,10, S_{\min} = 24,15$. The analysis of the data showed that the average daily change of price is $\Delta = 0,21$. So the state space is consisted of 22 elements. Statistical evaluations of non-central moments of the stock prices moving up and down are the following:

$$m_1^U = 2,2402;\ m_2^U = 11,6950;\ m_3^U = 85,7250;$$
$$m_1^D = 1,8321; m_2^D = 6,9676;\ m_3^D = 37,4190.$$

Corresponding parameters of approximating density function are calculated according to formulas given in section 3. The parameters are the following:

$$\mu_1^U = 0,4014;\ \mu_2^U = 1,2384;\ p_1^U = 0,2373;$$
$$\mu_1^D = 0,5381; \mu_2^D = 0,3358;\ p_1^D = 0,0088;$$

The dynamics of stock price movements was described in the event language presented in section 4. The developed software using the description as input data automatically generates all the possible states (the set of stock prices) and transition rates among them, calculates steady state probabilities of asset prices. The calculated probability distribution is given in the Table 1.

Table 1. Probability distribution of asset prices

State	24,15	24,36	24,57	24,78	24,99	25,2	25,41
Probability	0,0015	0,0017	0.0018	0.0018	0.028	0.0042	0.0079

State	25,41	25,62	25,83	26,04	26,25	26,46	26,67
Probability	0.0079	0.0154	0.0324	0.1609	0.1516	0.1187	0.1002

State	26,88	27,09	27,30	27,51	27,72	27,93	28,10
Probability	0.0860	0.0732	0.0613	0.0523	0.0453	0.0387	0.0423

The distribution can be used for evaluation of statistical measures of the stock prices and pricing derivative securities.

6 Conclusion

This paper gives a continuous time Markov chain model for asset dynamics. It allows solving of a large size problem. The model can be applied for pricing of option financial products working in continuous time and with countable number of possible values for the imbedded asset, which is always the case from the numerical point of view. The main interest of this model is that it works even when there are possibilities of arbitrage, i.e. the most frequent cases. Further research will be devoted for determining the risk neutral measure and pricing derivative securities based on the continuous time Markov chain model.

References

1. Black, F., Scholes, M.: The Pricing of Options and Corporate Liabilities. Journal of Political Economy 81, 637–659 (1973)
2. Cox, J., Rubinstein, M.: Options Markets. Prentice Hall, Englewood Cliffs (1985)
3. Madan, D.B., Seneta, E.: The VG model for share market returns. Journal of Business 63, 511–524 (1990)
4. Barndorff-Nielsen, O.E.: Normal inverse Gaussian distributions and the modelling of stock returns. Research Report no. 300, Department of Theoretical Statistics, Aarhus University (1995)
5. Carr, P., Geman, H., Madan, D.H., Yor, M.: The fine structure of asset returns: an empirical investigation. Journal of business 75, 305–332 (2002)
6. Eberlein, E., Keller, U.: Hyberbolic distributions in finance. Bernoulli 1, 281–299 (1995)
7. Grigelionis, B.: Proceses of Meixner type. Lithuanian Mathematics Journal 39(1), 33–41 (1999)
8. Whitt, W.: On Approximations for Queues, III: Mixtures of Exponential Distributions. AT&T Bell Labs Tech. Journal 63(1), 163–175 (1984)
9. Pranevičius, H., Valakevičius, E.: Numerical Models of Systems Specified b Markov Processes. Kaunas, Technologija (1996)

Foreign Exchange Rates Forecasting with a C-Ascending Least Squares Support Vector Regression Model

Lean Yu, Xun Zhang, and Shouyang Wang

Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190 China
{yulean,zhangxun,sywang}@amss.ac.cn
http://madis1.iss.ac.cn/

Abstract. In this paper, a modified least squares support vector regression (LSSVR) model, called C-ascending least squares support vector regression (C-ALSSVR), is proposed for foreign exchange rates forecasting. The generic idea of the proposed C-ALSSVR model is based on the prior knowledge that different data points often provide different information for modeling and more weights should be given to those data points containing more information. The C-ALSSVR can be obtained by a simple modification of the regularization parameter in LSSVR, whereby more weights are given to the recent least squares errors than the distant least squares errors while keeping the regularized terms in its original form. For verification purpose, the performance of the C-ALSSVR model is evaluated using three typical foreign exchange rates. Experimental results obtained demonstrated that the C-ALSSVR model is very promising tool in foreign exchange rates forecasting.

Keywords: Foreign exchange rates forecasting, least squares support vector regression, regularization parameter.

1 Introduction

The foreign exchange market is a nonlinear dynamic market with high volatility and thus foreign exchange rate series are inherently noisy, non-stationary and deterministically chaotic [1]. Due to its irregularity, foreign exchange rates forecasting is regarded as a rather challenging task. For traditional linear-based forecasting methods such as autoregressive integrated moving average (ARIMA) and exponential smoothing model (ESM) [2], it is extremely difficult to capture the irregularity because they are unable to capture subtle nonlinear patterns hidden in the foreign exchange rate series data.

To remedy the gap, many emerging nonlinear techniques, such as artificial neural networks (ANNs), were widely used in the foreign exchange rates forecasting and obtained good results relative to the traditional linear modeling techniques in the past decades. For example, De Matos [3] compared the strength of a multilayer feed-forward neural network (MLFNN) with that of a recurrent neural

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 606–615, 2009.

network (RNN) based on the forecasting of Japanese yen futures. Kuan and Liu [4] provided a comparative evaluation of the performance of MLFNN and a RNN on the prediction of an array of commonly traded exchange rates. In the article of Tenti [5], the RNN is directly applied to exchange rates forecasting. Hsu et al. [6] developed a clustering neural network (CNN) model to predict the direction of movements in the USD/DEM exchange rates. Their experimental results suggested that their proposed model achieved better forecasting performance relative to other indicators. In a more recent study by Leung et al. [7], the forecasting accuracy of MLFNN was compared with the general regression neural network (GRNN). The study showed that the GRNN possessed a greater forecasting strength relative to MLFNN with respect to a variety of currency exchange rates. Similarly, Chen and Leung [8] adopted an error correction neural network (ECNN) model to predict foreign exchange rates. Yu et al. [9] proposed an adaptive smoothing neural network (ASNN) model by adaptively adjusting error signals to predict foreign exchange rates and obtained good performance.

Although the ANN models achieve great success in foreign exchange rates forecasting, they still have some disadvantages in some practical applications. For example, ANN models often suffer from over-fitting problem in the case of when training was performed too long or where training examples are rare. Furthermore, local minimum problem often occurred due to the adoption of empirical risk minimization (ERM) principle in ANN learning. For this purpose, a competitive neural network learning model, called support vector machines (SVMs), was proposed by Vapnik and his colleagues in 1995 [10]. The SVM is a novel learning way to train polynomial neural networks based on the structural risk minimization (SRM) principle where seeks to minimize an upper bound of the generalization error rather than minimize the empirical error implemented in other neural networks. The generic idea is based on the fact that the generalization error is bounded by the sum of the empirical error and a confidence interval term that depends on the Vapnik-Chervonenkis (VC) dimension [11]. Using the SRM principle, the SVM will obtain a global optimum solution by adopting a suitable trade-off between the empirical error and the VC-confidence interval. Due to this distinct characteristic, the SVM has been widely applied to pattern classification and function approximation or regression estimation problems [12]. In terms of the classification and regression problems, the SVM can be categorized into support vector classification (SVC) and support vector regression (SVR). In this paper, we focus the SVR for foreign exchange rates forecasting.

In SVR, the problem is formulated by employing the so-called Vapnik's ε-insensitive loss function, taking the regression problem as an inequality constrained convex programming problem, more specifically a quadratic programming (QP) problem and using the Mercer condition for mapping from nonlinear feature space to the chosen kernel function [13]. Usually, this QP can lead to higher computational cost. For this purpose, least squares support vector machine (LSSVM), as a variant of SVM, tries to avoid the above shortcoming and obtain an analytical solution directly from solving a set of linear equations

instead of solving QP problem. In such a way, a least squares support vector regression (LSSVR) can be formulated by replacing Vapnik's ε-insensitive loss function with a least squares cost function corresponding to a form of ridge regression [13]. By introducing the least squares cost function, the LSSVM can be extended to solve nonlinear regression problems.

In the practical time series forecasting, many studies have shown that the relationship between input variables and output variable gradually changed over the time and recent data often contain more information than distant data. It is therefore advisable for us to give more weights to the recent containing more information. In view of this idea, an innovative approach is proposed by Tay and Cao [11] which used the ascending regularization parameter in the SVR to predict financial time series based on the work of Refenes and Bentz [14]. The ascending SVR is obtained by a simple modification of the regularization risk function in SVR, whereby the recent ε-insensitive errors are penalized more heavily than the distant ε-insensitive errors. The ascending SVR is reported to be very effective in financial time series forecasting.

This paper is motivated by the ascending SVR model, and generalizes the idea for LSSVR whereby more weights are given to the recent least squares errors than the distant lease squares errors in the regularization parameter. The primary objective of this paper is to propose a new forecasting paradigm called C-ascending LSSVR that can significantly reduce the computational cost and to improve the prediction capability of standard SVR as well as to examine whether the prior knowledge that recent data should provide more information than distant data can also be utilized by LSSVR in financial time series forecasting, especially for foreign exchange rates forecasting in this study.

The remainder of this paper is organized as follows. Section 2 overviews the formulation of least square support vector regression (LSSVR) model briefly. In Section 3, the formulation of the C-ascending LSSVR (C-ALSSVR) is presented in detail. For further illustration, three typical foreign exchange rates prediction experiments are conducted in Section 4. Finally, some concluding remarks are drawn in Section 5.

2 Least Squares Support Vector Regression (LSSVR)

Suppose that there is a give training set D of n data points $\{(x_i, y_i)\}_{i=1}^{n}$ with input data $x_i \in x \subseteq R^n$and output $y_i \in y \subseteq R$. An important idea of SVM is to map the input into a high-dimensional feature space F via a nonlinear mapping function $\varphi(x)$ to find an unknown function form f, which takes the following form

$$y = f(x; w, b) = w^T \varphi(x) + b \tag{1}$$

where $\varphi(\cdot)$ is a nonlinear mapping function, w_i is the ith weights, and b is a bias. In the least squares support vector regression, the following optimization problem is formulated

$$\min J = \frac{1}{2} w^T w + \frac{C}{2} \sum\nolimits_{i=1}^{n} e_i^2 \tag{2}$$

Subject to the following equality constraints

$$y_i = w^T\varphi(x_i) + b + e_i, \text{ for } i = 1, 2, \cdots, n. \tag{3}$$

where C is the regularization parameter, e_i is the ith approximation error between predicted and actual values. Combining (2) and (3), one can define a Lagrangian function

$$L(w, b, e_i; \alpha_i) = \frac{1}{2}w^T w + \frac{C}{2}\sum\nolimits_{i=1}^{n} e_i^2 - \sum\nolimits_{i=1}^{n} \alpha_i \left[w^T\varphi(x_i) + b + e_i - y_i\right] \tag{4}$$

where α_i is the ith Lagrangian multiplier. The optimal conditions are obtained by differentiating (4)

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \Rightarrow w = \sum\nolimits_{i=1}^{n} \alpha_i\varphi(x_i) \\ \dfrac{\partial L}{\partial b} = 0 \Rightarrow \sum\nolimits_{i=1}^{n} \alpha_i = 0 \\ \dfrac{\partial L}{\partial e} = 0 \Rightarrow e_i = \alpha_i/C \\ \dfrac{\partial L}{\partial \alpha_i} = 0 \Rightarrow w^T\varphi(x_i) + b + e_i - y_i = 0 \end{cases} \tag{5}$$

for $i = 1, 2, \cdots, n$. After elimination of e_i and w, the solution is given by the following set of linear equations

$$\begin{cases} \sum_{i,j=1}^{n} \alpha_i\varphi(x_i)^T\varphi(x_j) + b + (\alpha_i/C) - y_i = 0 \\ \sum_{i=1}^{n} \alpha_i = 0 \end{cases} \tag{6}$$

for $i, j = 1, 2, \cdots, n$. Using the Mercer condition, the kernel function can be defined as $K(x_i, x_j) = \varphi(x_i)^T\varphi(x_j)$ for $i, j = 1, 2, \cdots, n$. Typical kernel functions include linear kernel $K(x_i, x_j) = x_i^T x_j$, polynomial kernel $K(x_i, x_j) = \left(x_i^T x_j + 1\right)^d$, Gaussian kernel or RBF kernel $K(x_i, x_j) = \exp\left(-\|x_i - x_j\|^2 \Big/ \sigma^2\right)$, and MLP kernel $K(x_i, x_j) = \tanh\left(kx_i^T x_j + \theta\right)$ where d, σ, k and θ are kernel parameters, which are specified by users beforehand. Accordingly, (6) can be rewritten as

$$\begin{cases} \Omega\alpha + b\vec{1} = y \\ \vec{1}^T\alpha = 0 \end{cases} \tag{7}$$

where b is a scalar, Ω, α, y, and $\vec{1}$ are either matrix or vectors, which are defined, respectively,

$$\Omega = K(x_i, x_j) + (1/C)I, \tag{8}$$

$$\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n)^T, \tag{9}$$

$$y = (y_1, y_2, \cdots, y_n)^T, \tag{10}$$

$$\vec{1} = (1, 1, \cdots, 1)^T \tag{11}$$

where I is a unit matrix in (8). Equivalently, using the matrix form, the linear equations of (6) can be expressed by

$$\begin{bmatrix} \Omega & \vec{1} \\ \vec{1}^T & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ b \end{bmatrix} = \begin{bmatrix} y \\ 0 \end{bmatrix} \tag{12}$$

From (8), the Ω is positive definite, the solution of Lagrangian multiplier α can be obtained from (12), i.e.,

$$\alpha = \Omega^{-1}(y - b\vec{1}) \tag{13}$$

Substituting (13) into the second matrix equation in (12), we can obtain

$$b = \frac{\vec{1}^T \Omega^{-1} y}{\vec{1}^T \Omega^{-1} \vec{1}} \tag{14}$$

Here, since Ω is positive definite, Ω^{-1} is also positive definite and thus $\vec{1}^T \Omega^{-1} \vec{1} > 0$. Thus, b is always obtained. Substituting (14) into (13), α can be easily obtained. Accordingly the solution of w can be obtained from the first equation in (5). Using w and b, the function f shown in (1) can be determined.

The distinct advantages of LSSVR reflect the following two-fold. On the one hand, the optimal solution of (1) can be found by solving a set of linear equations instead of solving a quadratic programming (QP) problem which is used in standard SVR thus reducing the computational costs. On the other hand, relative to the standard SVR with the Vapnik's ε-insensitive loss function, there is no need to determine an additional ε accuracy parameter which is related to ε-insensitive loss function and reducing the chance of over-fitting.

3 *C*-Ascending Least Squares Support Vector Regression

As is known to many researchers in the field of machine learning, the regularization parameter C shown in (2) determines the trade-off between the regularized term and the tolerable empirical errors. With the increase of C, the relative importance of empirical errors will grow relative to the regularized term, and vice versa. Usually, in standard SVR and LSSVR, the empirical risk function has equal weight C to all ε-insensitive loss function [11] and least squares errors e_i ($i = 1, 2, \ldots, n$) between the predicted and actual values [13]. That is, the regularization parameter C is a constant or a fixed value.

However, many time series forecasting experiments (e.g., [11]) have shown that a fixed regularization parameter is unsuitable for some prediction tasks with some prior knowledge. Considering that different data might contain different information, more weights should be given to those data offering more information. In the case of LSSVR for financial time series forecasting, more weights should be given to the recent data taking the prior knowledge into account that recent data might offer more information than distant data. For this

purpose, the regularization parameter C should be replaced by a variable regularization parameter C_i to capture the variation of information containing in the time series data $\{(x_i, y_i)\}_{i=1}^n$. In terms of the prior knowledge that recent data might offer more information than distant data, the variable regularization parameter C_i should satisfy $C_i > C_{i-1} (i = 2, \ldots, n)$. Since the variable regularization parameters C_i will grow from the distant data points to the recent data points, C_i is called ascending regularization parameter which will give more weights on the more recent data points. In the practical applications, the form of C_i often depends on the prior knowledge we have. In the financial time series forecasting, two typical forms: linear form and exponential form [11] are often used.

For the linear ascending form, the number of training data points are often used to determine the value of C_i. A common linear form is shown as follows:

$$C_i = \frac{i}{n(n+1)/2} C = \frac{2i}{n(n+1)} C \tag{15}$$

where i is a time factor, n is the number of training data point, and C is a constant that needs tuning. Usually the more recent the training data point is, the larger the regularization parameter C_i is, according to (15). A distinct advantage of linear form is simplicity and easy to implementation. When the size of training data set is small and low computational cost is expected, it is a good choice.

For the exponential ascending form, another parameter r is introduced into the exponential function in order to control the rate of ascending, which is represented as

$$C_i = \frac{i}{1 + \exp(r - 2ri/n)} C \tag{16}$$

Exponential form could offer more ways to describe the changing of the information containing in the training data. When the size of training data set is large and the computational cost is not concerned, the exponential form can be adopted.

Besides the linear and exponential forms, other forms that can control the ascending rates can also be employed in the practical application.

Based on the ascending regularization parameter C_i, a new LSSVR called C-ascending LSSVR (C-ALSSVR) can be introduced. Similar to (2) and (3), the optimization problem of C-ALSSVR for time series prediction can be formulated as follows.

$$\begin{cases} \min \ J = \frac{1}{2} w^T w + \frac{C_i}{2} \sum_{i=1}^n e_i^2 \\ \text{s.t.} \quad y_i = w^T \varphi(x_i) + b + e_i, \text{ for } i = 1, 2, \cdots, n. \end{cases} \tag{17}$$

Using the Lagrangian theorem, the final solution is similar to (13) and (14). The only difference is the value of Ω due to the introduction of the variable regularization parameter C_i. In the case of C-ALSSVR, the value of Ω is calculated by

$$\Omega = K(x_i, x_j) + (1/C_i) I \tag{18}$$

According to (17) and (18), the LSSVR algorithm can still be used except the regularization parameter value C_i for every training data points is different, and

thus computation and simulation procedures of LSSVR should be utilized by a simple modification of regularization parameter from a fixed value C to a variable parameter C_i in terms of (17).

4 Experimental Results

In this section, three real-world foreign exchange rates are used to test the effectiveness of the proposed C-ascending LSSVR model. The data used here are monthly and are obtained from Pacific Exchange Rates Services, provided by Professor Werner Antweiler, University of British Columbia, Vancouver, Canada. They consist of the US dollar against each of the three currencies — British pounds (GBP), euros (EUR) and Japanese yen (JPY) studied in this paper. We take monthly data from January 1971 to December 2000 as in-sample (training periods) data sets (360 observations including 60 samples for cross-validations). We also take the data from January 2001 to November 2008 as out-of-sample (testing periods) data sets (95 observations), which is used to evaluate the good or bad performance of prediction based on some evaluation measurement. For evaluation, two typical indicators, normalized mean squared error ($NMSE$) [1] and directional statistics (D_{stat}) [1] are used. Given N pairs of the actual values (or targets, x_t) and predicted values ($\hat{x}_t$), the $NMSE$ which normalizes the MSE by dividing it through the variance of respective series can be defined as

$$NMSE = \frac{\sum_{t=1}^{N}(x_t - \hat{x}_t)^2}{\sum_{t=1}^{N}(x_t - \bar{x}_t)^2} = \frac{1}{\delta^2}\frac{1}{N}\sum_{t=1}^{N}(x_t - \hat{x}_t)^2 \tag{19}$$

where δ^2 is the estimated variance of the data and $\bar{x}_t$ the mean. Usually the $NMSE$ is only a level prediction evaluation criterion, which is not enough for financial forecasting. For this, the directional statistics (D_{stat}) is developed, which is defined by

$$D_{stat} = \frac{1}{N}\sum_{t=1}^{N} a_t \times 100\% \tag{20}$$

where a_t=1 if $(x_{t+1} - x_t)(\hat{x}_{t+1} - x_t) \geq 0$, and a_t=0 otherwise.

In addition, for comparison purpose, a reverse model relative to C-ALSSVR, called the C-descending LSSVR (C-DLSSVR), standard LSSVR without using the most recent 60 data points (LSSVR-60), and standard LSSVR (LSSVR) are used here. If the prediction performance of the C-DLSSVR and LSSVR-60 is worse than that of LSSVR, the prior knowledge that the recent training data points can offer more information for modeling than the distant training data points will be confirmed clearer.

In the experiment of C-ALSSVR, the Gaussian function is used as the kernel function and two ascending forms are examined. The regularization parameter C and kernel parameter σ are determined by the cross validation method. In particular, the validation set is used to choose the best combination of C, σ, and the optimal control rate r in the exponential ascending form. These values could

vary in different foreign exchange rates due to different characteristics of foreign exchange rates. The LS-SVMlab1.5 for solving the regression problem [15] is implemented in this experiment and the program is developed using Matlab language. For the LSSVR, the LSSVR-60, and the C-descending LSSVR which will put more weights on the more distant training data points, the same settings with C-ALSSVR are adopted.

Using the above experimental design, the corresponding computational results are reported in Tables 1 and 2 from the point of level prediction and direction prediction. In the two tables, a clear comparison of various methods for the three currencies is presented via $NMSE$ and D_{stat}. Generally speaking, the results obtained from the two tables also indicate that the prediction performance of the proposed C-ascending LSSVR model is better than those of the standard LSSVR, LSSVR-60 and C-DLSSVR models for the three main currencies.

Table 1. The $NMSE$ comparisons of different models for three foreign exchange rates

Models	GBP		EUR		JPY	
	$NMSE$	Rank	$NMSE$	Rank	$NMSE$	Rank
LSSVR	0.0238	3	0.0142	3	0.0981	3
LSSVR-60	0.0345	4	0.0204	4	0.1345	4
C-DLSSVR$_{Lin}$	0.0439	5	0.0289	5	0.1767	5
C-DLSSVR$_{Exp}$	0.0503	6	0.0366	6	0.2145	6
C-ALSSVR$_{Lin}$	0.0195	2	0.0093	2	0.0824	2
C-ALSSVR$_{Exp}$	0.0158	1	0.0065	1	0.0678	1

Table 2. The D_{stat} comparisons of different models for three foreign exchange rates

Models	GBP		EUR		JPY	
	$D_{stat}(\%)$	Rank	$D_{stat}(\%)$	Rank	$D_{stat}(\%)$	Rank
LSSVR	75.79	3	73.68	3	69.47	3
LSSVR-60	64.21	5	66.32	4	52.63	6
C-DLSSVR$_{Lin}$	65.26	4	61.05	5	61.11	4
C-DLSSVR$_{Exp}$	58.95	6	55.79	6	57.89	5
C-ALSSVR$_{Lin}$	78.95	2	80.00	2	71.58	2
C-ALSSVR$_{Exp}$	83.15	1	84.21	1	77.89	1

Focusing on the $NMSE$ indicator, our proposed C-ascending LSSVR model with exponential ascending form performs the best in all the cases, followed by the C-ascending LSSVR model with linear ascending form, standard LSSVR, LSSVR-60 model, and the C-descending LSSVR model is the worst. This indicates that the proposed C-ascending LSSVR model is more suitable for foreign exchange rates prediction than the standard LSSVR and C-descending LSSVR models. Interestingly, for the testing case of JPY, the $NMSEs$ of the all models are larger than those of the other two currencies. This might be because the JPY is more volatile than the other two currencies.

However, the low $NMSE$ does not necessarily mean that there is a high hit ratio for foreign exchange movement direction prediction. Thus the D_{stat} comparison is necessary for business practitioners. Focusing on D_{stat} of Table 2, we are not hard to find that the proposed C-ascending LSSVR model outperforms the other three models according to the ranking; furthermore, from the business practitioners' point of view, D_{stat} is more important than $NMSE$ because the former is an important decision criterion in foreign exchange trading. With reference to Table 2, the differences between the different models are very significant. For instance, for the EUR testing case, the D_{stat} for the C-DLSSVR model with exponential form is only 55.79%, for the LSSVR-60 method it is 66.32%, and the D_{stat} for the standard LSSVR model is 73.68%; while for the C-ALSSVR model with exponential form, D_{stat} reaches 84.21%. Furthermore, like $NMSE$ indicator, the proposed C-ALSSVR model with exponential form also performs the best in all the cases, followed by the C-ALSSVR model with linear form and standard LSSVR model, and the C-DLSSVR model with exponential form performs the worst. The main reasons are that in the foreign exchange rates forecasting the recent training data points are more important than the distant training data points. That is, the recent data could offer more information for modeling. By incorporating this prior knowledge into the LSSVR methods, the C-ALSSVR models are more effective in foreign exchange rates forecasting than the standard LSSVR models.

5 Concluding Remarks

In this study, a new least squares support vector regression (LSSVR) model, called C-ascending LSSVR (C-ALSSVR) model, is proposed for foreign exchange rates prediction. In terms of the empirical results, we can find that across different models for the test cases of three main currencies — British pounds (GBP), euros (EUR) and Japanese yen (JPY) — on the basis of different evaluation criteria, our C-ascending LSSVR models perform the best. In the presented forecasting cases, the $NMSE$ is the lowest and the D_{stat} is the highest, indicating that the proposed C-ascending LSSVR models can be used as a promising tool for foreign exchange rates prediction.

In addition, an interesting research direction is to explore more complex ascending functions which can closely follow the dynamics of foreign exchange rates series. Furthermore, the proposed C-ascending LSSVR models can be easily extended into other financial time series forecasting problems. We will look into these issues in the future research.

Acknowledgements

This work is partially supported by grants from the National Natural Science Foundation of China (NSFC No. 70601029, 70221001) and the Knowledge Innovation Program of the Chinese Academy of Sciences.

References

1. Yu, L., Wang, S.Y., Lai, K.K.: Foreign-Exchange-Rate Forecasting with Artificial Neural Networks. Springer, New York (2007)
2. Lai, K.K., Yu, L., Wang, S.Y., Huang, W.: Hybridizing Exponential Smoothing and Neural Network for Financial Time Series Predication. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3994, pp. 493–500. Springer, Heidelberg (2006)
3. De Matos, G.: Neural Networks for Forecasting Exchange Rate. M. Sc. Thesis. The University of Manitoba, Canada (1994)
4. Kuan, C.M., Liu, T.: Forecasting Exchange Rates Using Feedforward and Recurrent Neural Networks. Journal of Applied Econometrics 10, 347–364 (1995)
5. Tenti, P.: Forecasting Foreign Exchange Rates Using Recurrent Neural Networks. Applied Artificial Intelligence 10, 567–581 (1996)
6. Hsu, W., Hsu, L.S., Tenorio, M.F.: A Neural Network Procedure for Selecting Predictive Indicators in Currency Trading. In: Refenes, A.N. (ed.) Neural Networks in the Capital Markets, pp. 245–257. John Wiley and Sons, New York (1995)
7. Leung, M.T., Chen, A.S., Daouk, H.: Forecasting Exchange Rates Using General Regression Neural Networks. Computers & Operations Research 27, 1093–1110 (2000)
8. Chen, A.S., Leung, M.T.: Regression Neural Network for Error Correction in Foreign Exchange Rate Forecasting and Trading. Computers & Operations Research 31, 1049–1068 (2004)
9. Yu, L., Wang, S.Y., Lai, K.K.: Adaptive Smoothing Neural Networks in Foreign Exchange Rate Forecasting. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3516, pp. 523–530. Springer, Heidelberg (2005)
10. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
11. Tay, F.E.H., Cao, L.J.: Modified Support Vector Machines in Financial Time Series Forecasting. Neurocomputing 48, 847–861 (2005)
12. Scholkopf, B., Burges, C., Smola, A. (eds.): Advances in Kernel Methods – Support Vector Learning. MIT Press, Cambridge (1998)
13. Suykens, J.A.K., Lukas, L., Vandewalle, J.: Sparse Approximation Using Least Squares Support Vector Machines. In: Proceedings of The 2000 IEEE International Symposium on Circuits and Systems, vol. 2, pp. 757–760. IEEE Press, New York (2000)
14. Refenes, A.N., Bentz, Y., Bunn, D.W., Burgess, A.N., Zapranis, A.D.: Financial Time Series Modeling with Discounted Least Squares Back-Propagation. Neurocomputing 14, 123–138 (1997)
15. Pelckmans, K., Suykens, J.A.K., Van Gestel, T., De Brabanter, J., Lukas, L., Hamers, B., De Moor, B., Vandewalle, J.: LS-SVMlab: a Matlab/C Toolbox for Least Squares Support Vector Machines. Internal Report 02-44, ESAT-SISTA, K.U. Leuven, Leuven, Belgium (2002)

Multiple Criteria Quadratic Programming for Financial Distress Prediction of the Listed Manufacturing Companies

Ying Wang[1], Peng Zhang[1], Guangli Nie[1], and Yong Shi[1,2,*]

[1] Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing, 100190, China
[2] College of Information Science & Technology, University of Nebraska at Omaha, Omaha, NE 68182, USA
wangying.bj.cn@gmail.com, nczhang1999@163.com, sdungl@163.com, yshi@gucas.ac.cn

Abstract. Nowadays, how to effectively predict financial distress has become an important issue for companies, investors and many other user groups. The purpose of this paper is to apply the Multiple Criteria Quadratic Programming (MCQP) model to predict financial distress of the listed manufacturing companies. Firstly, we introduce the formulation of MCQP model. Then we use ten-folder cross validation to test the stability and accuracy of MCQP model on a real-life listed companies' financial ratios dataset. At last, we compare MCQP model with other two well-known models: Logistic Regression and SVM models. The experimental results show that MCQP is accurate and stable for predicting the financial distress of the listed manufacturing companies. Consequently, we can safely say that MCQP is capable of providing stable and credible results in predicting financial distress.

Keywords: Financial distress prediction, MCQP, Logistic, SVM.

1 Introduction

With the rapid development of capital market in recent years, making accurate financial distress prediction has become more and more important to many user groups, such as bank loan officers, investors, creditors, regulators and auditors [1]. The last few decades have witnessed a large body of research work on predicting finance distress from financial statement. As early as 1966, Beaver [2] discussed that the default prediction problem could be regarded as a problem of evaluating the probability of financial distress conditional upon the value of a specific financial ratio. In 1968, Altman [3] developed a Z-score bankruptcy prediction model based on five financial ratios using Multiple Discriminant Analysis (MDA). In 1980, Ohlson [4] applied the logistic regression model in bankruptcy prediction research. Unlike MDA, the logistic regression model does not based on normal distribution or the equality of covariance matrices of the two groups. He was followed by several other authors:

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 616–624, 2009.

Mensah [5], Casey and Bartczak [6] and Gentry at al [7]. From the late 1980s, the Artificial Intelligence (AI) or Machine Learning (ML) techniques were introduced to financial distress prediction studies [8, 9]. Most recently, Yu-Chiang Hu and Jake Ansell [10] applied the SVM model which based on convex quadratic programming to predict financial distress.

In recent years, researchers from Multiple Criteria Mathematical Programming (MCMP) are stepping into data mining field and propose many promising classification models. For instance, in 2001, Y. Shi [11] built up the Multiple Criteria Linear Programming (MCLP) and Multiple Criteria Quadratic Programming (MCQP) models which have received many attentions as their successful applications in finance, biology, medical insurance and many other social fields. The purpose of this paper is to apply the MCQP model to predict the financial distress of the listed manufacturing companies in China.

The rest of this paper is organized as follows: in Section 2, we introduce the MCQP model; in Section 3, we introduce the financial ratios dataset of the listed manufacturing companies; in Section 4, we test MCQP on this dataset using 10-folder cross-validation and compare its performance with two well-known models: Logistic Regression and SVM; in Section 5, we conclude our paper with some discussions.

2 Multiple Criteria Quadratic Programming (MCQP) Model

In this section, we will give a short introduction of MCQP model. Assume a two-group classification problem $\{G_1, G_2\}$. Given a training sample $Tr = \{G_1, G_2\}$, where n is the total number of records in the training sample. Each training instance $A_i (i = 1, ...n)$ has r attributes. A boundary scalar b is used to separate G_1 and G_2. Thus a vector $X = (x_1, x_2 ... x_n) \in R^r$ can be identified to establish the following linear inequality [12]:

$$\begin{aligned} A_i X < b &, \text{ some } A_i \in G_1 \\ A_i X \geq b &, \text{ some } A_i \in G_2 \end{aligned} \tag{1}$$

To formulate the criteria and complete constraints for data separation, some variables will be introduced. α_i is defined to measure the overlapping of two-group boundary for record A_i, that means if $A_i \in G_1$ but we misclassified it into G_2 or vice versa, there is a distance α_i and the value equals $|A_i X - b|$. Then β_i is defined to measure the distance of record A_i from its adjusted boundary b^*, that means if A_i is correctly classified, there is a distance β_i and the value equals $|A_i X - b^*|$, where $b^* = b + \alpha_i$ or $b^* = b - \alpha_i$. Suppose $f(\alpha)$ denotes for the relationship of all overlapping α_i while $g(\beta)$ denotes for the aggregation of all distances β_i. The final absolute catch rates depend on simultaneously minimizing $f(\alpha)$ and maximizing $g(\beta)$. By using the l_p norm to represent $f(\alpha)$ and l_q norm to represent $g(\beta)$ respectively, we get a generalized bi-criteria programming framework as follows:

$$\begin{aligned}
&\text{(Model 1)} \quad Minimize \left\|\alpha\right\|_p^p \ and \ maxmize \left\|\beta\right\|_q^q \\
&subject\ to: \\
&\quad A_i X - \alpha_i + \beta_i - b = 0,\ A_i \in G_1 \\
&\quad A_i X + \alpha_i - \beta_i - b = 0,\ A_i \in G_2 \\
&\quad \alpha_i, \beta_i \geq 0,\ i = 1, ..., n
\end{aligned} \tag{2}$$

Where A_i is given, X and b are unrestricted.

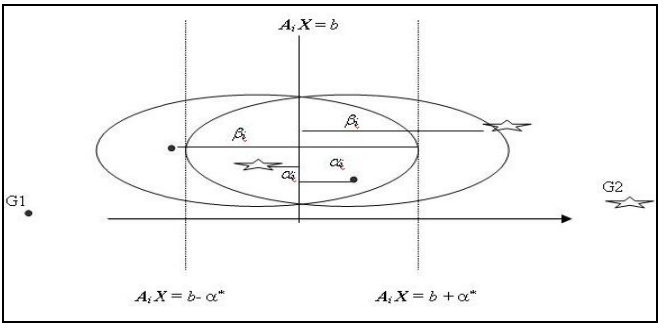


Fig. 1. Two-group classification model

All variables and their relationships are represented in Figure 1. There are two groups in Figure 1: black dot indicates G_1 data objects and star indicates G_2 data objects. There is one misclassified data object from each group if the boundary scalar b is used to classify these two groups, whereas adjusted boundaries $b - \alpha_i$ and $b + \alpha_i$ separate two groups without misclassification.

As far as Model 1 is considered, when setting p = q = 2 and combing the two objective functions into a single objective function by using $w_\alpha \geq 0$ and $w_\beta \geq 0$, we can get the MCQP model as follows:

$$\begin{aligned}
&\text{(Model 2)} \quad Minimize\ w_\alpha \sum_{i=1}^{n} \alpha_i^2 - w_\beta \sum_{i=1}^{n} \beta_i^2 \\
&subject\ to: \\
&\quad A_i x - \alpha_i + \beta_i - b = 0, \forall A_i \in G_1 \\
&\quad A_i x + \alpha_i - \beta_i - b = 0, \forall A_i \in G_2 \\
&\quad \alpha_i, \beta_i \geq 0,\ i = 1, ..., n
\end{aligned} \tag{3}$$

Model 2 is a non-convex problem, thus it is also a NP-hard problem. It is very difficult to get the global minimizer, especially for large problem. In order to solve (3) efficiently, we propose an algorithm, which converges to a local minimizer of (2).

In order to describe the algorithm in detail, we introduce some notation.

Let $\omega = (X, \alpha, \beta, b)$, $f_1(\omega) = w_\alpha \sum_{i=1}^{n} \alpha_i^2 - w_\beta \sum_{i=1}^{n} \beta_i^2$, and

$$\Omega = \left\{ \begin{array}{c} (X, \alpha, \beta, b) : A_i X - \alpha_i + \beta_i - b = 0, \forall i \in G_1, \\ A_i X + \alpha_i - \beta_i - b = 0, \forall i \in G_2, \\ \alpha_i \geq 0, \beta_i \geq 0, i = 1, \cdots, n \end{array} \right\}$$ be the feasible region of model 2.

Let $\chi_\Omega(\omega)$ be the index function of set Ω, i.e. $\chi_\Omega(\omega)$ is defined as follows

$$\chi_\Omega(\omega) = \begin{cases} 0, \omega \in \Omega, \\ +\infty, \omega \notin \Omega. \end{cases}$$

Then (3) is equivalent to the following problem

$$\min \ f_1(\omega) + \chi_\Omega(\omega) \tag{4}$$

Rewrite $f_1(\omega) + \chi_\Omega(\omega)$ as $f_1(\omega) + \chi_\Omega(\omega) = g_1(\omega) - h_1(\omega)$. Where $g_1(\omega) = \frac{1}{2}\rho \|\omega\|^2 + w_\alpha \sum_{i=1}^{n} \alpha_i^2 + \chi_\Omega(\omega), h_1(\omega) = \frac{1}{2}\rho \|\omega\|^2 + w_\beta \sum_{i=1}^{n} \beta_i^2$, $\rho > 0$ is a small positive number. Then $g_1(\omega)$ and $h_1(\omega)$ are convex functions. Apply the simplified DC algorithm [13] to problem (4), we get the solution as follow:

Algorithm 1. Given initial point $\omega^0 \in R^{3n+1}$ and parameter $\varepsilon > 0$ at each iteration $k \geq 1$, compute ω^{k+1} by solving the convex quadratic programming.

(Q^k) $\min \left\{ \frac{1}{2}\rho \|\omega\|^2 + \sum_{i=1}^{n} \alpha_i^2 - (h_1'(\omega^k), \omega), \ \omega \in \Omega \right\}$

The stopping criterion is $\|\omega^{k+1} - \omega^k\| \leq \varepsilon$. The sequence $\{w^k\}$ generated by Algorithm 1 converges to a local minimizer of (3).

3 Financial Ratios Dataset

The sample is selected from the manufacturing companies listed in the Shanghai Stock Exchange (SSE) and the Shenzhen Stock Exchange (SZSE). Financial states of these companies are categorized into two classes: healthy and distressed. Companies which are Specially Treated (ST) by China Securities Supervision and Management Committee (CSSMC) will be taken as the distressed ones, while those which are never specially treated by CSSMC will be seen as the healthy ones. 775 manufacturing companies are selected, of which 46 are ST companies and the remained 729 are healthy ones. For healthy companies, their financial statements in

the last year will be used, while for ST companies, their financial statements in the year before the special treated year[1] will be used.

In this paper, we use the financial statements between 2003 and 2007, then 40 ST companies are selected out of the total 46 ST companies. Due to the scarcity of ST companies, a matched-pair design is used to compose the examples. Each ST company is matched with two healthy companies randomly selected from the 729 healthy companies. By doing so, the sample is composed of 120 listed manufacturing companies, including 40 ST records and 80 healthy records.

According to the standards of summarization, measurability, and sensitivity, 24 financial rations which derived from the financial statements are calculated for the

Table 1. Descriptive statistics of the financial ratios (N=120)

Principle	Vari-ables	Financial ratios	Min.	Max.	Mean	StdDev
Liability	1	Current ratio	0.175	6.300	1.406	0.976
	2	Acid-test ratio	0.000	5.534	0.995	0.779
	3	Net cash flow from operating activities / Current liability	-10.54	1.976	0.05	1.066
	4	Current liability / Total liability	-0.988	2.629	0.914	0.269
Operational Efficiency	5	Turnover rate of account receivable	0.202	26.20	4.665	3.449
	6	Turnover rate of inventory	0.243	4738	50.79	431.9
	7	Turnover rate of total assets	0.019	5.49	0.917	0.719
	8	Cost of sales / Revenue	0.056	1.116	0.745	0.207
	9	Selling & distribution expense / Revenue	0.003	0.938	0.075	0.107
	10	G&A expence /Revenue	0.005	27.14	0.424	2.484
	11	Finance expence / Revenue	-0.006	13.93	0.166	1.273
Profitability	12	Asset Profit Ratio	-1.666	0.475	0.029	0.289
	13	Return on assets	-1.683	0.462	0.009	0.277
	14	Net profit margin	-28.66	0.391	-0.538	3.338
Growth	15	Revenue growth ratio	-0.778	2.993	0.226	0.466
	16	Net profit growth ratio	-523.5	11.47	-11.50	56.84
	17	Total assets growth ratio	-0.703	3.542	0.248	0.496
	18	Equity growth ratio	-19.53	3.829	0.068	2.455
	19	Gross profit growth ratio	-1.305	3.716	0.18	0.733
Structure	20	Asset-liability ratio	0.114	3.092	0.598	0.372
	21	Long-term liabilities /Total assets	-0.080	0.813	0.066	0.108
	22	Fixed asset, other assets & intangible assets / Total assets	0.000	0.798	0.387	0.167
Cash flow	23	Cash received from sales of goods or rendering services / Revenue	0.561	5.846	1.085	0.479
	24	Net cash flow from operating activities / Net profit	-3.871	10.81	0.652	1.295

[1] The Year before the Special Treated Year (YSTY) is defined as follows: suppose the time when a company is specially treated as the benchmark year t_0. If the company is specially treated among January to April, the YSTY is defined as two years before t_0. If the company is specially treated among May to December, YSTY is defined as the last year before t_0.

120 companies. These financial rations cover liability ratios, operational efficiency ratios, profitability ratios, growth ratios, structure ratios and cash flow ratios which are listed in Table 1.

4 Experiments

4.1 Empirical Study

Cross-validation is frequently used for estimating generalization error, model selection, experimental design evaluation, training exemplars selection, or pruning outliers. By definition, cross-validation is the practice of partitioning a sample of data into sub samples so that analysis is initially performed on a single sub sample, while further sub samples are retained "blind" in order for subsequent use in confirming and validating the initial analysis[14]. The basic idea is to set aside some of the data randomly for training a model, then the data remained will be used to test the performance of the model. In this paper, a ten-folder cross-validation is used to test MCQP's performance as shown in Algorithm 2. The data gathered is divided into two groups. The training group is composed of 20 ST records and 20 healthy records and the testing group is composed of 20 ST records and 60 healthy records. The process to select training and testing sets is described as follows: first, 20 ST records and 20 healthy records are randomly selected from the dataset. Then, they are combined to form a single training dataset, with the remained 20 ST records and 60 healthy records merged into a testing set.

Algorithm 2

Input: The data set $A = \{A_1, A_2, ..., A_n\}$, boundary b

Output: Training accuracy R_{tr} , testing accuracy R_{ts}

Begin

Repeat ten times

Step1. Generate the training set {TR} and testing set {TS} .

Step2. Sort all of the 24 attributes of X^* , the larger X^* , the more important of this attribute.

Step3. Calculate the score of each record A_i and get score array Score[i] = $A_i X^*$, for $\forall A_i \in A$.

Step4. Compare Score[i] with boundary b , if $A_i \in G_1$ with Score[i]< b , or $A_i \in G_2$ with Score[i]$\geq b$, then increase the number of correctly classified records N_{tr} in training set and N_{ts} in testing set.

Step5. Calculate the classification performance both on training set ($N_{tr} / |TR|$) and testing set ($N_{ts} / |TS|$) and return the training accuracy R_{tr} and testing accuracy R_{ts} .

End.

4.2 Experiment Results

Table 2 shows the ten-folder cross-validation result on the financial ratios dataset. The columns "ST" and "H" refer to the number of records that are correctly classified as

"ST" and "Healthy", respectively. The column "Accuracy" is calculated using correctly classified records divided by the total records in that class.

From Table 2, we can see that the average accuracies of 10 groups training sets is 98.00% on the ST companies and 100% on the healthy companies, while the average accuracies of 10 groups testing sets is 90.00% on the ST companies and 96.50% on the healthy companies. The results indicate that a good separation of the ST class and Healthy class is observed with MCQP model.

Table 2. Results of MCQP model on financial ratios dataset

Cross-validation	Training Set				Testing Set			
	ST	Accuracy	H	Accuracy	ST	Accuracy	H	Accuracy
DataSet 1	20	100.00%	20	100.00%	16	80.00%	59	98.33%
DataSet 2	20	100.00%	20	100.00%	18	90.00%	56	93.33%
DataSet 3	19	95.00%	20	100.00%	18	90.00%	52	86.67%
DataSet 4	19	95.00%	20	100.00%	20	100.00%	57	95.00%
DataSet 5	20	100.00%	20	100.00%	17	85.00%	60	100.00%
DataSet 6	20	100.00%	20	100.00%	17	85.00%	60	100.00%
DataSet 7	20	100.00%	20	100.00%	18	90.00%	59	98.33%
DataSet 8	19	90.00%	20	100.00%	19	95.00%	57	95.00%
DataSet 9	20	100.00%	20	100.00%	19	95.00%	60	100.00%
DataSet 10	20	100.00%	20	100.00%	18	90.00%	59	98.33%
Average		98.00%		100.00%		90.00%		96.50%

4.3 Comparison of MCQP with Logistic Regression and SVM

Table 3 exhibits the comparison results of MCQP, Logistic Regression and SVM. The first column lists the three algorithms. The second column is the recall rate of ST companies (more attention are paid to capture ST companies). And the third column is the accuracy of correctly classified companies (including both ST and healthy companies). It was found that the logistic regression model is slightly better than MCQP model on accuracy. However, MCQP model performs a little better on recall rate. Actually, recall rate is more important than accuracy as the misclassification of a real ST company will lead to a heavy loss. SVM achieves the least recall rate and accuracy. The reason may be that when doing classification, SVM chooses some marginal points as the delegates in each class, and it also thinks maximizing the distance of the delegates is maximizing the distance of each class. However, since our financial ratios dataset of the listed manufacturing companies probably obeys the Gaussian distribution, which means the points around the center is of the most important, choosing the marginal points as the delegates is not persuasive.

Table 3. Comparison of MCQP, Logistic, SVM

Algorithms	Testing records	
	recall	Accuracy
MCQP	90.00%	94.88%
Logistic Regression	87.50%	95.00%
SVM	82.50%	92.50%

5 Conclusion

How to effectively predict financial distress becomes an urgent need for bank loan officers, investors, creditors, regulators and auditors. As a promising data mining approach, Multiple Criteria Quadratic Programming (MCQP) method has been extensively applied in many business activities. In this paper, we introduced MCQP into predicting financial distress of China' listed manufacturing companies. At the beginning, we collected and built a real-life listed manufacturing companies' dataset, which has 120 records with 24 attributes. Then we tested MCQP on this dataset using ten-folder cross-validation. Finally, we compared MCQP with Logistic Regression and SVM. The experimental results show that MCQP is accurate and stable for predicting the financial distress, and moreover, MCQP is superior to Logistic Regression and SVM models in the sense of accuracy and average recall rate. Consequently, we can say that MCQP model is capable of predicting the financial distress of the listed manufacturing companies accurately and stably.

Acknowledgement

This research has been partially supported by a grant from National Natural Science Foundation of China (#70621001, #70531040, #70501030, #10601064, #70472074), National Natural Science Foundation of Beijing #9073020, 973 Project #2004CB720103, Ministry of Science and Technology, China and BHP Billiton Co., Australia.

References

1. Ko, P.C., Lin, P.C.: An evolution-based approach with modularized evaluations to forecast financial distress. Knowledge based systems 19, 84–91 (2006)
2. Beaver, W.H.: Financial ratios as predictors of failure. Journal of Accounting Research 4, 71–111 (1966)
3. Altman, E.I.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. Journal of Finance 23, 71–111 (1966)
4. Ohlson, J.A.: Financial ratios and the probabilistic prediction of bankruptcy. Journal of Accounting Research 18, 109–131 (1980)
5. Mensah, Y.M.: An examination of the stationarity of multivariate bankruptcy prediction models: A methodologi-cal study. Journal of Accounting Research 22, 380–395 (1984)
6. Casey, C., Bartczak, N.: Using operating cash flow data to predict financial distress: some extensions. Journal of Accounting Research 23, 384–401 (1985)
7. Gentry, J.A., Newbold, P., Whitford, D.T.: Classifying bankrupt firms with funds flow components. Journal of Accounting Research 23, 146–160 (1985)
8. Coats, P.K., Fant, L.F.: Recognizing financial distress patterns using a neural network tool. Financial Management 22, 142–155 (1993)
9. Zhang, G.P., Hu, M.Y., Patuwo, B.E., Indro, D.C.: Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. European Journal of Operational Research 116, 16–32 (1999)
10. Hu, Y.-C., Ansell, J.: Measuring retail company performance using credit scoring techniques. European Journal of Operational Research 183, 1595–1606 (2007)

11. Olson, D., Shi, Y.: Introduction to Business Data Mining. McGraw-Hill/Irwin (2007)
12. Fisher, R.A.: The Use of Multiple Measurements in Taxonomic Problems. Annals of Eugenics 7, 179–180 (1936)
13. An, L.T.H., Tao, P.D.: Solving a class of linearly constrained indefinite quadratic problem by D. C. algorithms. Journal of Global Optimization 11, 253–285 (1997)
14. Schneider, J.: Cross Validation, http://www.cs.cmu.edu/schneide/tut5/node42.html

Kernel Based Regularized Multiple Criteria Linear Programming Model

Yuehua Zhang[1,*], Peng Zhang[1], and Yong Shi[1,2]

[1] CAS Research Center on Fictitious Economy and Data Sciences,
Beijing 100080, China
[2] College of Information Science & Technology,
University of Nebraska at Omaha,
Omaha, NE 68182, USA
{zhangyuehua07,zhangpeng04}@mails.gucas.ac.cn
yshi@omaha.edu

Abstract. Although Regularized Multiple Criteria Linear Programming (RMCLP) model has shown its effectiveness in classification problems, its inherent drawback of linear formulation limits itself into only solving linear classification problems. To extend RMCLP into solving non-linear problems, in this paper, we propose a kernel based RMCLP model by using a form $w = \sum_{i=1}^{N} \beta_i \phi(x_i)$ to replace the original weight w in RMCLP model. Empirical studies on synthetic and real-life datasets demonstrate that our new model is capable to classify non-linear datasets. Moreover, comparisons to SVM and MCQP also exhibit the fact that our new model is superior to other non-linear models in classification problems.

Keywords: MCLP, RMCLP, classification, kernel function.

1 Introduction

Data mining is defined as "The nontrivial extraction of implicit, previously unknown, and potentially useful information from data"[1]. Traditionally, data mining uses machine learning, statistical and visualization techniques to discover and present knowledge in a form which is easily comprehensible to humans. From the aspect of methodology, data mining can be performed through association, classification, clustering, prediction, sequential patterns, and similar time sequences. [2].Classification is one of the most important parts of data mining. Classification generally includes three steps: the first step is to build a model by using a given data which is predetermined. The next step is to test the model. The last step is to use the model to predict unlabeled data if the model accuracy is high enough.

Recent years have witnessed a large body of research work on mining useful knowledge by multiple criteria mathematical programming method where various of classification models have been proposed and received great success in business intelligence [1]. All these models are created mainly by adapting the objective functions of the original multiple criteria linear programming (MCLP) model [4] to improve

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 625–632, 2009.

MCLP's accuracy and stability, and lots of research works have exhibited their powerful ability to classify different kinds of real-life data. Among all these models, the most recent Regularized Multiple Criteria Linear Programming (RMCLP) model [3] which is created by adding two regularized objective functions into the original MCLP model, has been theoretically demonstrated that it is stable in finding the global optimal solution. The experiment results show that RMCLP model is an effective classification model.

However, RMCLP is inadequate to classify non-linear data sets. To overcome this shortage, in this paper, we add kernel function into the original RMCLP model to enable it to solve non-linear classification problems.

The remained of this paper is organized as follows. In section 2, we introduce the kernel algorithm. In section 3, we give a short review of MCLP and RMCLP model. In section 4, we formulate our new kernel based RMCLP model. In Section 5, we perform experiments on a synthetic dataset. In Section 6, we use two real-life data sets to compare our model with two other non-linear models: MCQP and SVM. In the last section, we conclude our paper with discussions.

2 Kernel Algorithms

Linear classification problem is the most basic situation in the entire classification problems. To classify the non-linear problems, we need to construct non-linear mapping functions. For example, SVM uses a non-linear map φ to map a input vector x to a higher dimensional space Z (which is also called as the feature space), then it constructs Optimal Hyper plane in the feature space Z. Assuming the non-linear map is $\varphi: R^d \to Z$, we transform the vector x in the original space to a new vector $z = \phi(x)$ in the feature space Z. Considering the input vector is $x = (x^1, x^2, ..., x^d)$ with weights $w_i = \alpha_i y_i$, we can get the SVM discrimination function as follows:

$$y = \text{sgn}(\sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b). \tag{1}$$

The parameter $K(x_i, x)$ is the non-linear transformation of the i-th support vector. Figure 1 shows the framework of SVM in the sense of neural networks:

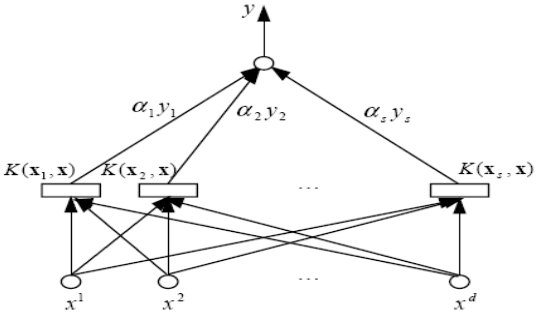


Fig. 1. The framework of SVM

In the discrimination function:

$$\begin{aligned} y &= \mathrm{sgn}(\sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b) \\ &= \mathrm{sgn}(\sum_{i=1}^{N} \beta_i \cdot \phi(x_i) \cdot \phi(x) + b) \\ &= \mathrm{sgn}(w \cdot \phi(x) + b) \end{aligned} \tag{2}$$

Then we set w= $\sum_{i=1}^{N} \beta_i \phi(x_i)$ where ϕ denotes a higher dimensional feature map associated with the nonlinear kernel and β_i denotes the variables. By the Representer theorem (Scholkopf and Smola, 2002), we indeed know that the optimal solution has the following form:[6]

$$K(x_i, x) = \langle \phi(x_i), \phi(x) \rangle \tag{3}$$

The most common kernel functions are listed as follows [8]:

(1) Radial Basis Kernel Function:

$$K(x_i, x) = \exp(-c \left\| x - x_i \right\|^2 \Big/ \sigma^2), \tag{4}$$

where c is a constant, σ is variance;

(2) Polynomial Kernel Function:

$$K(x_i, x) = ((x \cdot x_i) + c)^d, \tag{5}$$

where d is a positive real number, $c \geq 0$;

(3) Sigmoid Kernel Function:

$$K(x_i, x) = \tanh(b(x \cdot x_1) + c), \tag{6}$$

Where *b>0, c<0*.

3 MCLP and RMCLP Models

3.1 The Formulation of MCLP Model

Suppose each evaluated target x_i is described by n attributes (or variables). Consider l targets where data observation of the *i-th* target is $x_i = (x_{i1},...,x_{in})^T$, for $i = 1...l$. In linear discriminate analysis, the purpose is to determine the optimal coefficients (or weights) for the attributes, denoted by $w = (w_1,...,w_n)^T$ and a boundary value (scalar) b to separate two predetermined classes: G (Good) and B (Bad); that is

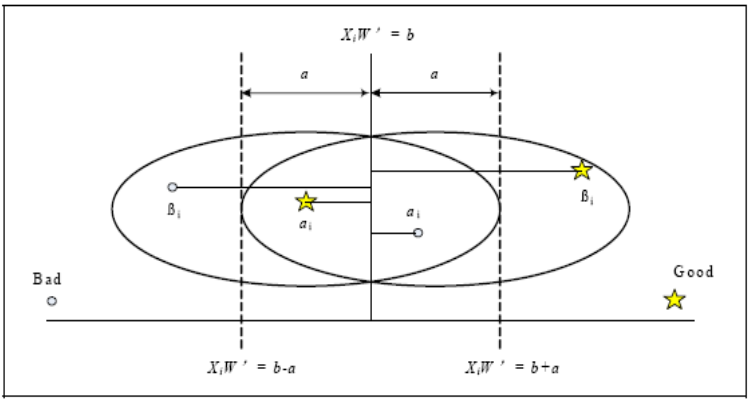


Fig. 2. Overlapping of two-class linear discriminate analysis

To measure the separation of Good and Bad, α_i is defined to be the overlapping of the two-class boundary for case x_i . Then, α is denoted as the max overlapping of the two-class boundary for all cases $x_i (\alpha_i < \alpha)$. β_i is also defined to be the distance of case x_i from its boundary, while β is the minimum distance of all cases x_i to the boundary $(\beta_i < \beta)$. To find the compromise solution of MMD (maximize the minimum distances) and MSD (minimizing the sum of the deviations) for data separation, we want to minimize the sum of α_i and maximize the sum of β_i simultaneously, as follows:

$$
\begin{aligned}
&\textit{Minimize} \quad C\sum_{i=1}^{l}\alpha_i - \sum_{i=1}^{l}\beta_i \\
&\textit{s.t.}: \\
&\quad x_{11}w_1 + \ldots + x_{1n}w_n = b + \alpha_1 + \beta_1, \quad \textit{for} \quad x_1 \in B, \\
&\quad x_{l1}w_1 + \ldots + x_{\ln}w_n = b + \alpha_l + \beta_l, \quad \textit{for} \quad x_1 \in G, \\
&\quad x_{l1}w_1 + \ldots + x_{\ln}w_n = b + \alpha_l + \beta_l, \quad \textit{for} \quad x_1 \in G, \\
&\quad \alpha_i \geq 0, \quad i = 1, \ldots l, \\
&\quad \beta_i \geq 0, \quad i = 1, \ldots l, \\
&\quad w_i \in R^n.
\end{aligned}
\tag{7}
$$

As we discussed above, the inherent drawback of linear formulation makes MCLP can only solve linear classification problems.

3.2 RMCLP Model

Lots of empirical studies have shown that MCLP is a powerful tool for classification. However, there is no theoretical work on whether MCLP always can find an optimal

solution under different kinds of training samples. To go over this difficulty, recently, Shi et.al [5] proposed a RMCLP model by adding two regularized items $\frac{1}{2}x^T Hx$ and $\frac{1}{2}\alpha^T Q\alpha$ on MCLP as follows:

$$\begin{aligned} &Minimize \frac{1}{2}w^T Hw + \frac{1}{2}\alpha^T Q\alpha + d^T\alpha - c^T\beta \\ &s.t.: \\ &\quad x_{i1}w_1 + ... + x_{in}w_n = b + \alpha_i - \beta_i, \forall\, x_i \in G1; \\ &\quad x_{i1}w_1 + ... + x_{in}w_n = b - \alpha_i + \beta_i, \forall x_i \in G2; \\ &\quad \alpha_i, \beta_i \geq 0. \end{aligned} \tag{8}$$

where $H \in R^{r*r}, Q \in R^{n*n}$ are symmetric positive definite matrices. $d^T, c^T \in R^n$ .The RMCLP model is a convex quadratic program. Theoretically studies [5] have shown that RMCLP can always find a global optimal solution.

4 Kernel Based RMCLP Model

In order to solve the non-linear classification problem, we add kernel function into the RMLCP model. As discussed above, we use $w = \sum_{i=1}^{N} \beta_i \phi(x_i)$ to replace the original weight *w* in RMCLP model. By letting $\beta_i = \lambda_i y_i$, we get $w = \sum_{i=1}^{N} \lambda_i y_i \phi(x_i)$ and the new RMCLP model can be formulated as follows:

$$\begin{aligned} &Minimize \ \ \frac{1}{2}w^T Hw + \frac{1}{2}\alpha^T Q\alpha + d^T\alpha - c^T\beta \\ &s.t.: \\ &\quad \lambda_1 y_1(\phi(x_1)\cdot\phi(x_i)) + ... + \lambda_n y_n(\phi(x_n)\cdot\phi(x_i)) = b + \alpha_i - \beta_i, \forall\, x_i \in G1; \\ &\quad \lambda_1 y_1(\phi(x_1)\cdot\phi(x_i)) + ... + \lambda_n y_n(\phi(x_n)\cdot\phi(x_i)) = b - \alpha_i = \beta_i, \forall\, x_i \in G2; \\ &\quad \alpha_i, \beta_i \geq 0. \\ &\quad C \leq \lambda_i \leq E; \end{aligned} \tag{9}$$

Where $H \in R^{r*r}, Q \in R^{n*n}$ are symmetric positive definite matrices. $d^T, c^T \in R^n$. Besides, we add the last constrain $C \leq \lambda_i \leq E$ when we compare the model with SVM. By letting $K(x_i, x_j) = \phi(x_i)\cdot\phi(x_j)$, we get the kernel based RMCLP model in Equ. (10).

$$
\begin{aligned}
&\textit{Minimize}\ \ \frac{1}{2}w^T Hw+\frac{1}{2}\alpha^T Q\alpha+d^T\alpha-c^T\beta\\
&s.t.:\\
&\lambda_1 y_1 K(x_1,x_i)+\ldots+\lambda_n y_n K(x_n,x_i)=b+\alpha_i-\beta_i,\forall\, x_i\in G1;\\
&\lambda_1 y_1 K(x_1,x_i)+\ldots+\lambda_n y_n K(x_n,x_i)=b-\alpha_i=\beta_i,\forall\, x_i\in G2;\\
&\alpha_i,\beta_i\geq 0.\\
&C\leq\lambda_i\leq E;
\end{aligned}
\tag{10}
$$

5 Testing on Synthetic Dataset

In order to investigate the performance of our new model in classifying non-linear problems, we design a two-group non-linear dataset as shown in Fig. 3:

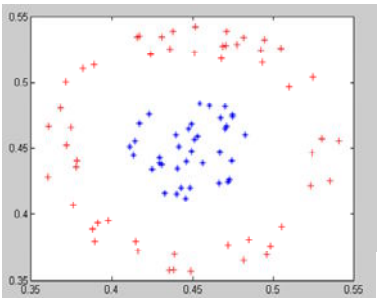


Fig. 3. Original dataset

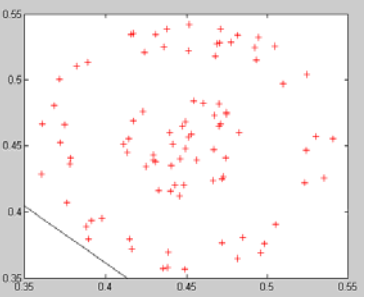


Fig. 4. Result with original RMCLP

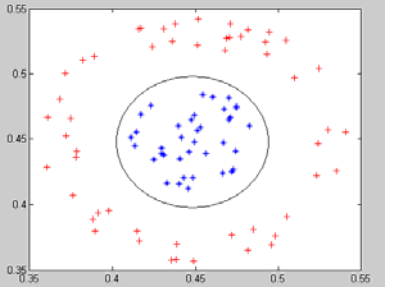


Fig. 5. Result with Polynomial kernel RMCLP

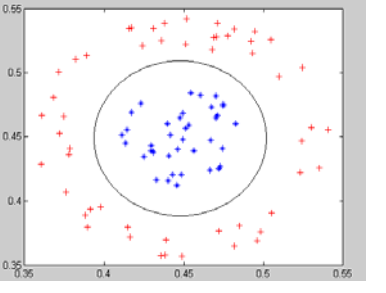


Fig. 6. Result with RBF kernel RMCLP

Our synthetic dataset contains 90 records, with 38 records in Group 1 (the blue points) and 52 records in Group 2 (the red points). Since the number of instances in the two groups is imbalanced, RMCLP model puts all records into Group 2 which is

bigger than Group 1. It can reach the best accuracy among all linear classifiers. Fig. 5 and Fig. 6 show the results of the kernel based RMCLP (with Polynomial kernel and RBF kernel respectively). The two groups are perfectly separated.

6 Experimental Results

In this section, we use two public UCI datasets to compare the performance of Kernel based RMCLP with two other methods: SVM and MCQP (a quadratic formulation of MCLP). Here we only use linear kernel to SVM.

Before building models, we scale each feature into the range [0, 1]. For the *Australian* dataset, we randomly divide it into two parts: training set with 200 records and testing set with 490 records. For *Heart* dataset, we also randomly split it into two parts: training set with 100 records and testing set with 170 records.

In every training process, parameters in Kernel based RMCLP are selected in some discrete sets in order to get the best accuracy. We select the parameter E from the set [0.1 1 16 128 1024 10000], C from the set [-∞ 0], and γ from [0.001 0.01 0.1 1 16 128 1024].[2] When getting the model for every C, E and γ, we test the performances on the training sets. Then, we fix the C, E and γ set which achieves the highest accuracies to predict the test sets and record the accuracies in Table 1 and Table 2.

Table 1. Test On Australian Dataset

Classification Algorithms	**Training(200 records)** Training Accuracy	**Testing (490 records)** Testing Accuracy
MCQP	83.5%	84.49%
SVM	85%	84.55%
RMCLP	85.5%	81.84%
K-RMCLP	90.5%	87.14%

Table 2. Test On Heart Dataset

Classification Algorithms	**Training(100 records)** Training Accuracy	**Testing (170 records)** Testing Accuracy
MCQP	84%	83.53%
SVM	84%	82.35%
RMCLP	79%	81.18%
K-RMCLP	90%	85.88%

In the table above, we can observe that RBF Kernel based RMCLP performances better than the other two algorithms. And the prediction accuracy improvement of our new model has a strong relationship with the linear-separable degree of the training sample. Through these experiments, we find out that the Kernel based RMCLP is a competitive method in non-linear classification.

7 Conclusions

In this paper, we add the kernel function into the Regularized Multiple Criteria Linear Programming (RMCLP) model to deal with the non-linear classification problems. By the *Representer Theorem*, we know that the optimal solution has the following form: $K(x_i, x) = \langle \phi(x_i), \phi(x) \rangle$. After replacing w in RMCLP model with $w = \sum_{i=1}^{N} \beta_i \phi(x_i)$, we upgrade the original RMLCP model into tackling the non-linear classification problems. Experimental results on both synthetic and real-life datasets show that our model is effective in classifying non-linear datasets. In the future work, we will study the parameters' effect of the kernel functions in our new kernel based RMCLP model.

References

1. Olson, D., Shi, Y.: Introduction to Business Data Mining. McGraw-Hill/Irwin (2007)
2. Zhang, Z., Zhang, D., Tian, Y., Shi, Y.: Kernel Based Multiple Criteria Linear Program
3. Frawley, W., Piatetsky-Shapiro, G., Matheus, C.: Knowledge Discovery in Databases: An Overview. AI Magazine, 213–228 (Fall 1992)
4. Shi, Y.: Multiple criteria and multiple constraint levels linear programming: concepts, techniques and applications. World Scientific Pub. Co Inc., New Jersey (2001)
5. Shi, Y., Tian, Y., Chen, X., Zhang, P.: A Regularized Multiple Criteria Linear Program for Classification. In: ICDM Workshops 2007, pp. 253–258 (2007)
6. Chapelle, O., Sindhwani, V.: Optimization Techniques for Semi-Supervised Support Vector Machines. Journal of Machine Learning Research 9, 203–233 (2008)
7. Zhang, P., Tian, Y., Zhang, Z., Li, X., Shi, Y.: Supportive instances for Regularized Multiple Criteria Linear Programming Classification
8. Deng, N., Tian, Y.: New Approach in Data Mining – Support Vector Machine. Science Press, Beijing (2004)

Retail Exposures Credit Scoring Models for Chinese Commercial Banks

Yihan Yang[1], Guangli Nie[2], and Lingling Zhang[1,2,*]

[1] School of Management, Graduate University of Chinese Academy of Sciences, Beijing 100190, China
[2] Chinese Academy of Sciences Research Center on Fictitious Economy and Data Science, Beijing 100190, China
nkyangyihan@yahoo.com.cn, sdungl@163.com, zhangll@gucas.ac.cn

Abstract. This paper firstly discussed several credit scoring models and their development history, then designed the target system of individual credit scoring with individual housing loans data of a stated-owned commercial bank and logistic method, and established an individual credit scoring model including testing. Finally, the paper discussed the application of the individual credit scoring model in consumer credit domain, and brought forward corresponding conclusions and policies.

Keywords: Credit Scoring; Consumer Credit; Logistic Regression.

1 Introduction

With the rapid advancement of our society in recent years, people's consumption and investment concept has changed greatly. The individual consumption loan has become an important financing channel for personal consumption. Individual consumption credit business, such as housing credit, automobile credit has greatly developed. However, with the rapid development of individual consumption credit in Chinese commercial banks, default events of individual consumption credit business happened a lot, and the level of consumption credit risk assessment needs to be improved immediately. Therefore, study on the individual consumption credit risk has important theoretical value and practical significance.

The New Basel Capital Accord not only built the minimum capital adequacy ratio which covers the sources of credit risk, market risk, and operational risk, supervision and inspection, and market constraints, but also put forward Standard Approaches and IRB (Internal Rating-Based Approaches) to measure credit risk. It pointed out that banks with full conditions should implement IRB and estimated customers' PD (Probability of Default) by building models with historical data [5]. Compared with the old Capital Accord of 1998, the New Capital Accord took the estimated value of commercial banks' internal rating system as input parameters for the capital calculation, which was a major innovation. These formulas are based on modern risk management technology, involving a large number of mathematical statistics, as well as quantitative risk analysis, which is of great help to strengthen the internal risk management.

* Corresponding author.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 633–642, 2009.

2 Customer Credit Rating and Its Development

From the development history of international commercial banking, we could see that the customer credit rating of commercial banks has evolved from Expert System, Credit Scoring, to the Probability of Default Model. Due to the relatively short operating history of China's commercial banks and their lack of experience in data analysis, most commercial banks still widely use the more traditional Credit Scoring method.

2.1 Expert System

Expert System is a traditional method of credit analysis which relies on credit experts' professional knowledge, skills and experience, uses a variety of specialized analysis tools, analyzes key elements and then makes a comprehensive assessment of credit risk based on subjective judgments. Table 1 summarizes some influential indicators

Table 1. Main Qualitative Indicators

CAMEL	Capital Adequacy
	Asset Quality
	Management
	Earnings
	Liquidity
5C	Character
	Capacity
	Capital
	Collateral
	Condition
5P	Personal
	Purpose
	Payment
	Protection
	Perspective
3F	Management Factor
	Financial Factor
	Economic Factor
CAMPARI	Character
	Ability
	Margin
	Purpose
	Amount
	Repayment
	Insurance
6A	Economic Aspects
	Technical Aspects
	Managerial Aspects
	Organizational Aspects
	Commercial Aspects
	Financial Aspects

both at home and abroad, mainly CAMEL system, 5C elements, CAMPARI elements, 5P elements, 3F elements, 6A elements, and so on.

2.2 Credit Scoring

Credit scoring model uses observable characteristic variables of the borrower, and calculates a numerical value (score) to represent the debtor's credit risk and classify borrowers into different risk levels. For individual customers, observable characteristic variables include income, assets, age, occupation, as well as place of residence, and so on. The key of credit scoring model is to choose characteristic variables and determine their respective weight. At present, the most widely used score models include Linear Probability Model, Logit Model, Probit Model, Decision Tree Method, ANN(Artificial Neural Networks), as well as SVM(Support Vector Machine).

Credit scoring began with Beaver's single-variable analysis of 79 bankrupt companies [1]; Altman put forward the Z score model and ZETA score model based on multivariate statistics [3]; Matin, Ohlson and Wigintonfor for the first time used Logit model to analyze the enterprise bankruptcy [6]; Katz et al. used discriminant analysis in credit scoring study; Ou and Penman adopted Probit model to predict corporate bankruptcy. In the application of artificial intelligence, Lee et al. made empirical analysis on credit scoring samples with two decision tree methods as CART and MARS [9]; Gestel et al., Min & Lee, Liu Min, Lin Chengde adopted support vector machine method in bankruptcy prediction and credit assessment, proving that the machine learning performance was superior to the traditional method [14].

This article focused on the application of Logit model,. Westgaard & Wijst used the enterprise's micro-economic information, integrated the financial indicators, and estimated the default probability of the portfolio in retail banking with Logit model. The method was used in the credit risk management of Norwegian business sector, achieving very good results [4]. Lanine and Vennet analyzed characteristics of banks inclining to bankruptcy with Logit model and trait recognition method, targeted at the confusion and crisis of Russian banks in the 1990s [10].

2.3 Probability of Default Model

Since the 1990s, a number of models that could directly calculate the default probability emerged, including Moody's RiskCalc model and Credit Monitor, KPMG's Risk-neutral Pricing Model and the Mortality Model.

Compared with the traditional Expert System and Credit Scoring method, the Probability of Default Model could directly estimate the customer's probability of default, and therefore had a high requirement for historical data, requiring commercial banks to establish a consistent and clear-cut definition of default, and to accumulate historical data of at least five years. According to the status quo of China's banking industry, commercial banks should combine default probability model with the traditional credit scoring, expert system, which could help raise the level of credit risk assessment.

3 The Process of Modeling

This paper designed the target system of individual credit scoring with individual housing loans data of a stated-owned commercial bank for credit scoring model verification. If we apply the framework to other credit products, such as car loans, personal consumption loans, and so on, we just need change the relative elements and maintain the overall framework. Flow chart of the model is shown in Fig. 1 below:

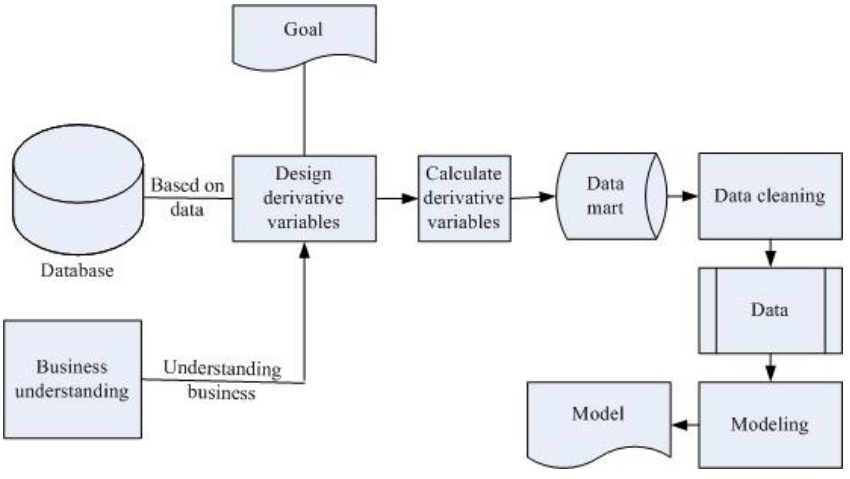


Fig. 1. Flow Chart of the Model

3.1 Establishing the Indicator System

The dependent variable for Logistic regression is discrete, and the predicted value of the model is output in form of probability [15]. Similar to the discriminant analysis, Logistic regression can deal with the issue of classification, while the difference is that Logistic regression requires neither normal distribution nor the equal variance assumption [9]. Personal credit data rarely meet with these two assumptions, so Logistic regression is more suitable for credit scoring models. The regression model built with Logistic distribution curve is also known as Logit model. The equation for logic model is as follows:

$$Ln(\frac{P}{1-P}) = \beta_0 + \beta_1 X_1 + \cdots\cdots + \beta_n X_n \tag{1}$$

We established the indicator system and listed quantitative indicators of the individual housing loan credit scoring model as Table 2 shows.

3.2 Data Collection

According to the definition of "normal loan" and "non-performing loan", we collect all individual housing loan samples covering all the indicators. Data in observation

period generate variable X; performance period comes after the observation period; we watch the performance of samples in this period, and generate variable Y.

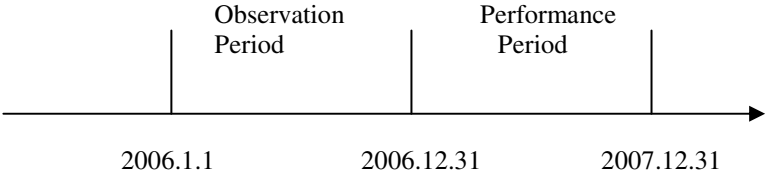


Fig. 2. Observation period & performance period

Data processing in observation period: According to the data available, we choose the observation period from 2006.1.1 to 2006.12.31, which is a whole year, preventing some holidays from impacting the model, and X_i is from data in this period.

We choose the performance period from 2007.1.1 to 2007.12.31, and determine the customer credit according to the customer's performance in this period.

3.3 Data Cleaning

According to the data available and China's actual situation, we identify the model's main factors X_i and the dependent variable Y (i = 1 2 ... ... .8). Main factors and data cleaning methods are shown in table 2.

Table 2. Main factors data cleaning

Explanatory Variables X_i	Cleaning Methods
Repayment Ratio (the amount of repayment / loans) X_1	
Down payment ratio (the amount of down payment /loans) X_2	We clean out those customers whose accounts were settled before 2006
Loan percentage X_3	If a customer has over one record, we choose the last record.
Loan period X_4	

Table 2. *(Continued)*

Non-repayment period ratio(non-repayment periods/total periods) X_5	For a record who has over one due bill, we choose the sum; for a record whose number of nor-repayment periods is 0, we clean it out.
Asset X_6	
Gender X_7	This indicator is generated in SAS system according to customers' ID number
Age X_8	Ibid

The number of cumulative overdue times accumulated in SAS system is shown in Fig. 3 below:

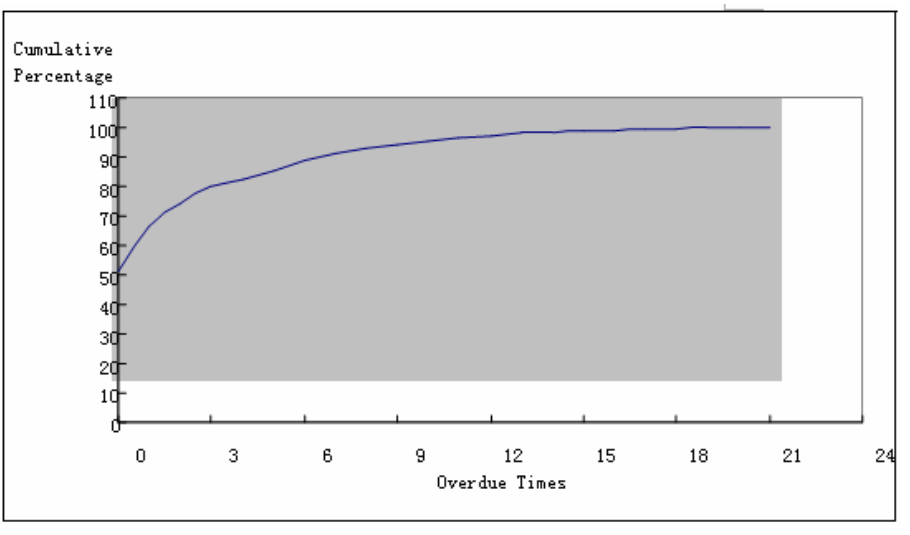


Fig. 3. Chart of cumulative times

As can be seen from the chart, when the cumulative overdue times are over 7, the overdue frequency changes very little, that's to say, the possibility for those customers to change into "good guys" is very small, so we take those whose cumulative overdue frequence is ≥ 7 as bad customers.

Y=1 if bad customers(cumulative overdue frequence ≥ 7 in 2007)
Y=0 if good customers (cumulative overdue frequence <7 in 2007)
From the chart, we could see that

$$\text{Credit Score}=1-\frac{1}{1+\exp(\sum_{i=1}^{N}-\beta_i X_i)} \tag{2}$$

X_{i} is the value of the explanatory variable I; β_i is the corresponding weight; N is the number of explanatory variables. The Score here is the ultimate value and the probability of being a good customer in this model. Therefore, the higher the value, the better the result is.

3.4 Logit Regression

We receive a total of 1317 records after data cleaning, and then we build a model with these 1317 customers and make corresponding parameter test.

Table 3. Result of logit regression

Independent Variable	Coefficient Estimation	Significance Test (Wald Test)	Goodness of Fit Test
Age	0.0759	0.0004	HL test is not significant (0.8425). The fit data R^2 is 0.9175, which means the fitness is good.
Total Assets(RMB)	-9.38E-07	0.0326	
Repayment Ratio	-0.9711	0.0075	
Down payment ratio	-426.7	0.0024	
Loan period	-0.0168	0.0002	
Loan quota	0.0155	0.0022	
Non-repayment period ratio	-2.4466	0.0066	

The relevant information of Logistic regression shows that the fitness is good. Hosmer-Lemeshow test (HL test) is not significant, R^2=0.9175 shows that the accuracy of the model is higher. Gender statistics are not significant and doesn't enter the model. The results of "Association of Predicted Probabilities and Observed Responses" show the correlation between probability forecasting and observed dependent variables. The Percent Concordant is 94.7%, and the Percent Discordant is 3.8%, which shows that there is a strong correlation between the predicted and observed values at the current level, and the regression model has a strong predicting ability.

3.5 Model Diagnosis

We operate a linear diagnosis on the model, and the result is shown in Table 4 below.

Table 4. Linear diagnosis

Independent Variable	Freedom	Tolerance	VIF
Age	1	0.9808	1.01957
Total Assets(RMB)	1	0.93513	1.06937
Repayment Ratio	1	0.98616	1.01403
Down payment ratio	1	0.70581	1.41682
Loan period	1	0.57696	1.73322
Loan quota	1	0.66951	1.49362
Non-repayment period ratio	1	0.56939	1.75627
Age	1	0.55633	1.7975

Usually we set VIF=10 or Tolerance=0.1 as threshold value. When VIF<10, or Tolerance>0.1,we think the variables selected have no obvious collinearity. The VIF of variable selected in this model is less than 5, so the model has no collinearity problem.

3.6 Model Explanation

(1)The estimated coefficient of the age factor is positive, having a negative correlation with the final score value. This is because customers in the database are aged 21-58. Customers with lower ages (generally 25-40 years old) are the main force in society, whose bright future and better income guarantee a lower loan default rate. Customers aged 45-58 are close to retirement, so the default rate is relatively higher. The result here happens to coincide with the personal credit score standard of China Construction Bank. The bank gives customers aged of 36-50 a credit score of 6, while those aged 50 and above a credit score of 4 (China Construction Bank Web site). This shows that above a certain limit of age, the older the customer, the higher the default rate.

(2)The estimated coefficient of the age factor is negative, having a positive correlation with the final score value. The more the assets are, the lower the default rate is. It is worth noting that the estimated coefficient of this indicator is relatively small in this model, and the weight appears to be low. Considering the reality that China's personal property declaration system is not perfect, and there is a certain degree of hidden income, we argue that the actual "monthly income" indicator would be of greater importance to personal credit than that in this paper.

(3) The estimated coefficients of Repayment Ratio, Down payment ratio, and Non-repayment period ratio are negative, having a positive correlation with the final score value. The estimated coefficient of the loan quota factor is positive, having a negative

correlation with the final score value, which shows that the higher the loan quota is, the larger the loan amount, and thus the bigger the probability of default; this is in line with the real business. It is worth noting that the estimated coefficient of the loan period factor is negative. According to general analysis of business, loans with longer duration will have more uncertainty, leading to greater likelihood of default, that is, the loan period should have had a positive correlation with the default rate. But the model here gives a negative coefficient. One explanation could be that the length of the loan agreement is influenced by the owner's risk attitude. Cautious customers are inclined to have loans with longer periods. Most Chinese customers of housing loans are risk avoiding, and their default probability is relatively low, so the loan period has a negative correlation with the default probability. This result shows that in China's current data accumulation, because of all the banks' scrutiny, customers who can get loans of longer periods generally have good credit.

4 Conclusions

Credit risk is the most dangerous one in financial industry. The construction of evaluation model of individual credit is urgently needed. Based on this, this article discussed the method of evaluating customer credit. In combination with the theory of personal credit risk, we choose Logistic regression with data of individual housing loan from some Chinese Commercial Bank, and make theoretical research and empirical analysis on the evaluation of individual credit risk management in Chinese Commercial Banks. The model based on Logistic regression shows that there are six indicators that greatly affect individual credit evaluation as follows: age, assets, loan quota, Repayment Ratio, Down payment ratio, and Non-repayment period ratio.

Through comparison between conclusions at home and abroad, it could be concluded that the model is right and effective, and the empirical result is consistent with the real experience. However, in the process of constructing the model, some basic information of customers which is very important to the construction of the model, such as education background, professions, marriage status, etc. is not taken into consideration due to bad quality of data in commercial bank. Therefore, we strongly suggest that China should establish a united basic database of individual credit information to obtain comprehensive data and mark the client credit more accurately. Because the samples in this article are consumption credit data taken from a typical Chinese commercial bank, the finally constructed individual credit scoring model can be a good reference for other commercial banks to evaluate individual consumption credit.

Acknowledgements

This research has been partially supported by a grant from National Natural Science Foundation of China (#70621001, #70531040, #70501030, #70472074), Beijing Natural Science Foundation (#9073020).

References

1. Beaver, W.: Financial rations are predictors of failure. Journal of Accounting Research (4), Suppl. 71–111 (1966)
2. Altman, E., Haldeman, N.P.: ZETA analysis: A new model to identify bankruptcy risk of corporations. Journal of Banking and Finance (1), 29–54 (1977)
3. Altman, E., Narayanan, P.: An International Survey of Business Failure Classification Models. Financial Markets, Institutions and Instruments 6(2) (1997)
4. Westgaard, S., Wijst, N.: Default probabilities in a corporate bank portfolio: A logistic model approach. European Journal of Operational Research (135), 338–349 (2001)
5. Basle Committee. Basle Committee on Banking Supervision. The New Basel Capital Accord. Bank for International Settlements, Basle (April 2003)
6. Wiginton, J.C.: A note on the comparison of logit and discriminant models of consumer credit behavior. Financial Quant. Anal. (15), 757–770 (1980)
7. Press, J., Wilson, S.: Choosing between Logistic regression and discriminant analysis. Journal of American Statistical Association 73(7), 699–705 (1978)
8. Scott, D.F., Martin, J.D.: Industry Influence on Financial Structure. Financial Management, 67–73 (Spring 1975)
9. Shin, K.S., Lee, T.S., Kim, H.J.: An application of support vector machines in bankruptcy prediction model. Expert Systems with Applications (28), 127–135 (2005)
10. Lanine, G., Vennet, R.V.: Failure prediction in the Russian bank sector with logit and trait recognition models. Expert Systems with Applications (30), 463–478 (2006)
11. Laitinen, E.K., Laitinen, T.: Bankruptcy prediction: application of the Taylor's expansion in logistic regression. Int. Rev. Financial Anal. 9(4), 327–349 (2000)
12. Flagg, J.C., Giroux, G.A., Wiggins, C.E.: Predicting corporate bankruptcy using failing firms. Rev. Financial Econ. (1), 67–78 (1991)
13. Kay, O.W., Warde, A., Martens, L.: Social differentiation and the market for eating out in the UK. Int. J. Hosp. Manage. 19(2), 173–190 (2000)
14. Gestel, T.V., Baesens, B., Suykens, J.A.K., Poel, D.V., Baestaens, D.E., Willekens, M.: Bayesian kernel based classification for financial distress detection. European Journal of Operational Research (2005)
15. Cox, D.R., Snell, E.J.: Analysis of Binary Data. Chapman & Hall, London (1989)

The Impact of Financial Crisis of 2007-2008 on Crude Oil Price

Xun Zhang, Lean Yu, and Shouyang Wang*

Institute of Systems Science, Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing 100190, China
{zhangxun,leanyu,sywang}@amss.ac.cn

Abstract. For better estimation of the impact of extreme events on crude oil price volatility, an EMD-based event analysis approach is proposed. In this method, the time series to be analyzed is first decomposed into several intrinsic modes with different time scales from fine-to-coarse and an average trend. The decomposed modes respectively capture the fluctuations caused by the extreme event or other factors during the analyzed period. The total impact of an extreme event is included in only one or several dominant modes, but other modes provide valuable information for subsequent factors. The effects of financial crisis of 2007-2008 to crude oil price are analyzed through this method and empirical results reveal that the EMD-based event analysis method provides a feasible solution to estimating the impact of extreme events on crude oil prices.

Keywords: Crude Oil Price, Financial Crisis, Event Analysis, Empirical Mode Decomposition.

1 Introduction

The financial crisis that began on 9 August 2007 ranks among the most serious economic events affecting the global economy since the Great Depression of the 1930s. The crisis has also affected the oil market and contributed to the collapse of oil prices in 2008. The closing price of West Texas Intermediate crude oil price, which rose to an all-time high of $145.31 per barrel on 3 July 2008, was driven down to $41 per barrel on December 5, 2008, a drop of 72% in five months, mainly due to the slowdown in economic growth in the United States and other parts of the world. In fact, the variations of crude oil price are strongly related to extreme events such as wars and economic recessions [1][2]. Some literatures have spawned to utilize the effects of historical extreme events in crude oil price modeling and forecasting. For example, TEI@I methodology, a systematic crude oil price forecasting methods incorporating the effects of extreme events have been constructed and its empirical results have shown its superiority [3][4]. Others methods based on historical information of events include rough-set refining text mining, pattern matching methods and so on [5][6].

* Corresponding author. Tel.: 86-10-62651375, Fax: 86-10-62621324

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 643–652, 2009.

However, the fundamental of incorporating historical events in forecasting is to analyze the effects of events. Traditional event evaluation methods mainly include two kinds: intervention analysis and event study approach. Intervention analysis is one of the most rigorous statistical modeling techniques, which is used to test whether a postulated event caused a change in the time series and, if so, what are the magnitude and the nature of the change [7][8]. This method is only applicable to linear and stationary time series because essentially it is based on linear time series model. Likewise, the event study method has been a standard analysis tool for assessing the financial or economic impact of specific unanticipated events in economics, accounting and finance [9]. It assumes the efficient market hypothesis (EMH) must be satisfied. But whether the EMH holds for crude oil markets is an open question.

In summary, both the two traditional methods consider a time series as the sum of a normal evolution part and an exceptional shock, and then they implement the "divide and conquer" strategy to separately modeling the two components. But as addressed by Huang et al. [10][11], although a complex time series can be treated as sums of several simple oscillation patterns, generally the number of these patterns is more than two. Therefore, a feasible decomposition method which can extract the inherent oscillations from the time series could provide more information for event evaluation. Therefore, a competitive decomposition algorithm — empirical mode decomposition (EMD), is applied to study the behaviors of crude oil price during the recent financial crisis.

In the innovative EMD-based event analysis approach, the original time series is first decomposed into some independent oscillations, and then the concrete implications of each oscillation are identified. Usually, the dominant oscillations are purely caused by the extreme event of interest and they are used to approximate the pattern and magnitude of the change. At the same time, other oscillations provide useful information about changes triggered by the event at different time scales, such as the effects of the event to short-term fluctuations and the long term trend. The EMD-based method has some advantages over the intervention analysis and event study approach. Firstly, it is suitable for nonlinear and non-stationary data. Secondly, it provides a multiscale framework to analyze the impact of the extreme event, including both direct impact and indirect impact.

The rest of this study is organized as follows. Section 2 briefly describes the EMD algorithm and the EMD-based event analysis approach. The effects of financial crisis to crude oil price are evaluated through this method in Section 3. In Section 4, Concluding remarks are given.

2 Methodology Formulation

2.1 Empirical Mode Decomposition

Empirical mode decomposition (EMD) is a promising nonlinear, non-stationary data processing method proposed by Huang et al. (1998) [10]. It considers the real time series as fast oscillations superimposed on slow oscillations. Those oscillations are approximated by "intrinsic mode functions" (here after IMF, or mode). An IMF must satisfy the following two conditions: 1) the numbers of extrema and zero-crossings

are the same, or differ at the most by one; 2) they are symmetric with respect to local zero mean.

A sifting process is designed to extract IMFs level by level. First, the IMF with the highest frequency riding on the lower frequency part of the data is extracted, and then the IMF with the next highest frequency is extracted from the differences between the data and the extracted IMF. The iterations continue until no IMF is contained in the residual. The overall sifting procedure for a time series x(t)is described as follows.

1) Initialize: set $r_0(t)=x(t)$, $i=$;,
2) Extract the i^{th} IMF:

2.1) Initialize: set $d_0(t)=r_{i-1}(t)$, $k=1$;
2.2) Identify all the maxima and minima of $d_{k-1}(t)$;
2.3) Generate the upper and lower envelopes, $e_{min}(t)$ and $e_{max}(t)$, of $d_{k-1}(t)$, with cubic spline interpolation.
2.4) Calculate the point-by-point mean, $m(t)$, from upper and lower envelopes:

$$m(t)= (e_{min}(t) + e_{max}(t))/2 \quad (1)$$

2.5) Extract the mean from the $d_{k-1}(t)$ and define the difference of $d_{k-1}(t)$ and $m(t)$ as $d_k(t)$:

$$d_k(t)= d_{k-1}(t) - m(t) \quad (2)$$

2.6) Check the properties of $d_k(t)$. If it is an IMF, denote $d_k(t)$ as the i^{th} IMF: $c_i(t)$; else, set $k=k+1$ and go back to 2.2);

3) Define $r_{i+1}(t)= r_i(t)- c_i(t)$;
4) Check whether the following two stopping criteria are satisfied: (1) the number of extrema included in $r_{i+1}(t)$ is smaller than 3; (2) the amplitudes of $r_{i+1}(t)$ are far smaller than the amplitudes of $r_i(t)$ at each point.

If the stopping criteria are not satisfied, set $i=i+1$ and go back to 2) to extract another IMF; else the sifting process is completed, and the final $r_{i+1}(t)$ is the average trend of $x(t)$. $x(t)$ can be represented as the sum of IMFs with the residue.

In practice, it is an improvement of EMD, ensemble EMD (EEMD) method, are widely used [12][13]. The EEMD is designed to overcome the mode mixing problem when intermittency exists in data.

2.2 EMD-Based Event Analysis

The overall process of applying EMD to event analysis is described as follows:

Step1: Determining data frequency and analysis window. The first step is selecting analyzed data according to the event of interest. Then the data is divided into two sub periods: estimation window and event window. The estimation window is defined as periods without the effects of the event, and the event window as periods that include the effects of the event.

Step2: Decomposing data by EMD. After the preliminary understanding of the event and identification of data, the time series is decomposed into several IMFs.

For data with intermittencies, EEMD is better than EMD for its ability to avoid the scale mixing.

Step3: Analyzing intrinsic modes. The first task in this step is to find the mode which sketches the overall change made by the extreme event in the analysis window. As mentioned before, each IMF has a concrete implication, representing a meaningful component of the original time series. Generally the effect of the extreme event is represented by one or sum of a few IMFs. This IMF or the sum of these IMF is treated as a main mode of the time series. Since the noises and long term trends contained in original time series are cleared away, the main mode gives a clear evaluation for the pattern and magnitude of change made by the extreme event in the event window. However, effects of some events, which themselves contain superimposed signals belonging to different scales, may be decomposed into several IMFs. In such a situation, those IMFs should be summed up into a component. Summing up more than one IMF is called "composition".

Besides the main mode, the event may also influence analyzed time series in other scales. For example, uncertainties brought by the event often make the analyzed indicator exhibit quick and small fluctuations. This can be tested by comparing the energy (square of the amplitude) of the IMF. Furthermore, the Hilbert spectrum can tell the differences in energy-frequency distribution between the event and estimation windows.

Finally, conclusions are drawn on the basis of analysis in accordance with steps 1) to 3). The pattern and magnitude of change made by the event are summarized. Then economic explanations are given.

3 Estimating the Effect of 2008 Financial Crisis on Crude Oil Price Volatility

3.1 The Data

The price series provided in this section include two benchmark daily time series for crude oil: West Texas Intermediate spot price (WTI) and European Brent spot price (Brent). The time period is divided into two sub periods: March 24, 2006 – August 8, 2007 as the estimation window and August 9, 2007 – December 16, 2008 as the event window. Each sub period contains 342 data points. All data are from Energy Information Administration, United States. We used the date as the dividing time line for these two periods when the large French bank BNP Paribas temporarily halted redemptions from three of its funds. Although a complete chronology of the crisis might start in February 2007 when several large subprime mortgage lenders started to report losses. The August 9, 2007 is recognized as the real trigger [14].

3.2 Decomposition by EEMD

EEMD is applied to decompose both WTI and Brent crude oil prices. An ensemble with 100 members is used, and the white noises added in each ensemble member are generated randomly from a normal distribution with standard deviation of 0.2. The stopping criterion for each sifting process is the same as that described in Section 2.1. Both the data series are decomposed into 8 IMFs plus 1 residue. Fig.1 is the visualization.

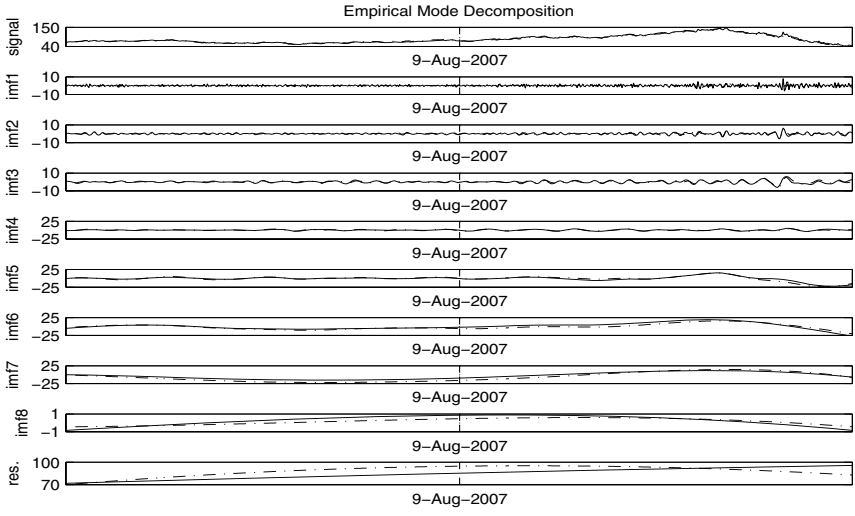


Fig. 1. The IMFs and residue for the WTI *(solid line)* and Brent *(dotted line)* crude oil prices. The first panel is the visualization of original time series.

In order to understand the features of price volatilities in different time scales, the values of IMFs in the estimation and event window are individually statistically analyzed. Three kinds of measures are calculated: mean period, correlation coefficients, and variance percentage of all the IMFs. Since the frequency and the amplitude of an IMF may change continuously with time, the periods are not constant. The mean period is defined as the value derived by dividing the total number of points by the number of peaks for each IMF. Correlation coefficients are used to measure the relationship between the individual component and the original time series. Variances of each IMF's percent of all IMFs are used to explain the contribution of each IMF to the total volatility of the observed data. The statistical measures are presented in Table 1 and Table 2.

Table 1. Measures of IMFs and the residue for the WTI and Brent daily crude oil price in the analysis window

	WTI / Brent (analysis window)		
	Mean Period	Correlation Coefficient	Variance percentage
IMF1	2.9 / 3.0	0.07 / 0.06	0.84 / 0.43
IMF2	6.6 / 6.3	0.08 / 0.06	0.57 / 0.39
IMF3	12.7 / 14.3	0.17 / 0.09	1.09 / 0.95
IMF4	34.2 / 34.2	0.08 / 0.18	1.66 / 1.09
IMF5	114 / 114	0.85 / 0.80	31.50 / 32.25
IMF6	171 / 171	0.96 / 0.95	43.61 / 25.83
IMF7	342 / 342	0.81 / 0.68	17.02 / 34.20
IMF8	342 / 342	0.24 / 0.46	0.11 / 0.03
Residue		0.08 / 0.30	3.60 / 4.84

Table 2. Measures of IMFs and the residue for the WTI and Brent daily crude oil price in the event window

	WTI / Brent (event window)		
	Mean Period	Correlation Coefficient	Variance percentage
IMF1	2.8 / 2.8	0.08 /0.09	0.76 / 0.35
IMF2	6.6 / 6.2	0.10 / 0.11	0.49 / 0.23
IMF3	13.2 / 12.2	0.20 / 0.16	0.85 / 0.39
IMF4	24.4 / 28.5	0.34 / 0.37	1.64 / 1.32
IMF5	57 / 68.4	0.41 / 0.52	4.30 / 4.58
IMF6	342 / 171	0.80 / 0.66	24.50 / 17.07
IMF7		0.66 / 0.46	40.89 / 33.36
IMF8		-0.33 / 0.02	0.46 / 0.07
Residue		-0.25 / -0.14	26.13 / 42.65

It can be seen that there is no significant difference between the two price series except for the residue in event window. The high frequency IMFs, including IMF1 to IMF3, represent small fluctuations and exhibit similar behavior in both the two sub periods. They not only exhibit very low correlation coefficients with the observed data but also account for less than 1.1% of total variance. This means that these IMFs do not have serious effect on crude oil price. IMF1 captures the fluctuations with a cycle of 3 days, that is, the impact which is eliminated after 3 days on average. Similarly, IMF2 captures fluctuations over 1 week and IMF3 captures fluctuations over 2 and a half week. All the amplitudes of high frequency IMFs are less than $8, which means high frequency fluctuations in the price series are no larger than $8.

The impact of this financial crisis on crude oil price mainly exerts on low frequency IMFs. For IMF4 and IMF5, the mean periods in event window are shorter than those in the estimation window. For IMF4 to IMF6, the amplitudes in event window are much higher than those in the estimation window.

The IMF6 and IMF7 are two dominant modes during the whole time range. Variances of IMF6 and IMF7 rank highly among all the IMFs. Correlation coefficients between the two IMFs and the price series also reach a high level. The IMF8 and residue satisfy the trend definition given by [15]. Therefore they are essentially the long-term trend.

3.3 Analyzing the Modes

Dominant mode: the pattern of major impacts. Since the IMF6 and IMF7 are identified as dominant mode during the event window, they are used to outline the pattern of major impact of the crisis. The sum of the IMF6 and IMF7 in event window are normalized to [0,1] and plotted in Fig.2. The dominant modes for WTI and Brent, almost perfectly match the shape of the original time series. We use this distance to measure the total impact of the financial crisis on crude oil prices. More specifically, by computing the distance of the local maximum in July 2008 and local minimum in the last day of the event window, we got the conclusion that the financial crisis have driven the WTI price downward $65 and Brent price downward $63 until December 16, 2008. By analyzing the dominant mode, of which the cycle does consist with the event window, rather than the original prices, the pattern and magnitude of the impact are very clear.

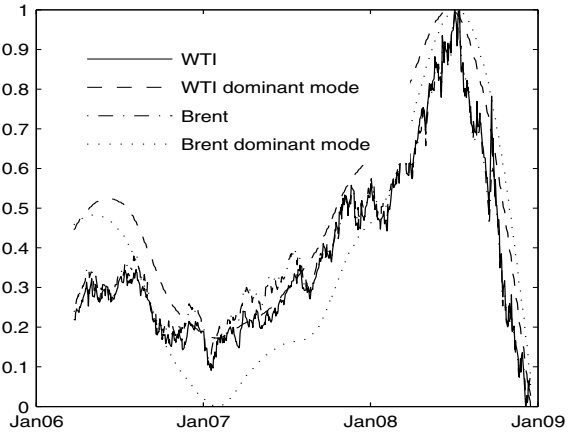


Fig. 2. The normalized prices and dominant mode derived from EEMD

High frequency IMFs: the total effect is near zero. The most high frequency mode, IMF1, captures most of the large and sudden jumps of the prices. The large changes of price series correspond to large amplitudes of IMF1. This means the period of the large jump is very short: a sharp rise (descent) always follows a sharp descent (rise) quickly. Thus the markets have the mean-reversion trend in the short-term.

However, in the whole event window, the total effects of these high frequency IMFs are near zero. This can be proved by a fine-to-coarse reconstruction. That is, using *t*-test to identify from which IMF c_i the mean of the sum of c_1 to c_i significantly departs from zero (null hypothesis). It shows that the mean of the reconstruction from IMF1 to IMF4 does not significantly depart from zero but the reconstruction from IMF1 to IMF5 does. Therefore, the sum of IMF1 to IMF4 does not have any long term effects on the prices and can be neglected if only the total impact of the event on the price series is of concern. This is another reason why we can take only IMF6 and IMF7 as the dominant IMFs representing the main change of prices during the event window.

The Spectrum analysis: the crisis amplifies the volatilities. In this analysis, we try to find whether the financial crisis amplified the crude oil price volatilities. Two methods, Hilbert spectrum analysis and t-test method are used. The EMD method, combined with Hilbert transform, is called Hilbert-Huang transform (HHT) since it is proposed by Huang et al. (1998). It presents the data clearly in a time-frequency-energy space. Fig. 3 shows the Hilbert spectrum of WTI and Brent crude oil prices. The vertical axis is the normalized instantaneous frequency and the grayscale of the contours represents the energy.

Note that at each time point, the energy is mainly distributed at low frequency. It is not strange since the dominant mode is the low frequency IMF6 and IMF7. Looking at the high frequency region, the energy clusters in the event windows are slightly denser than those in the estimation window. At the same time, the energy in the event windows is higher than that in the estimation window To visualize it more clearly, we

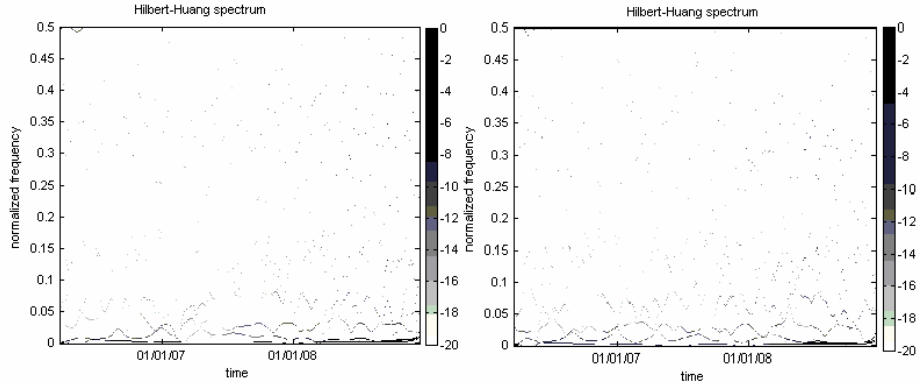


Fig. 3. The Hilbert spectrum of IMFs *(left: WTI; right: Brent)*

calculate the marginal energy density at time t, in normalized instantaneous frequency range [0.6, 1]. Intuitive results are shown in Fig. 4. It is very clear that the energy in the event window is much higher than that in the estimation window, especially the time the price drop sharply.

Then a t-test is implemented to test whether the amplitudes of IMF1 in the event window are significantly larger than those in the estimation window. The result does support the hypothesis. Therefore, we get the conclusion that the crisis actually increases the volatility of the crude oil markets in the event window.

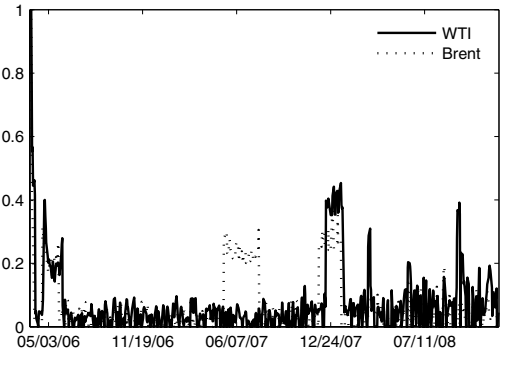


Fig. 4. The normalized prices and dominant mode derived from EEMD

4 Conclusions

In this paper, we first present an EMD-based event analysis method to estimate the impact of extreme events on crude oil price volatility, and then the effects of financial crisis of 2007-2008 to crude oil price are analyzed. The characteristics of the crisis to crude oil price can be summarized as below:

The crisis had several types of impacts on crude oil prices: (i) high frequency fluctuations with small amplitudes; (ii) a large shock as a delta impulse, the time span of which is consistent with the event.

The sum of small fluctuations is near zero and does not have a long-term effect on crude oil prices. Analysts who address only the major impact of the event could ignore these high frequency fluctuations. But the volatilities of prices are increased during the financial crisis. This finding is similar with Ferderer (1996), which discovers the recessions of economic activities increase oil price volatility.

The large shock caused by the financial crisis, is represented by low frequency IMFs with long mean period and accounts for most of the variations in the event window. Through the numerical calculation, the financial crisis have driven the crude oil price downward more than $60 until December, 2008, roughly 40% of the highest price, except for the effects of speculative activities and other impact factors.

Acknowledgments. This work is partially supported by grants from the National Natural Science Foundation of China (NSFC No. 70601029, 70221001).

References

1. Yang, C.W., Hwang, M.J., Huang, B.N.: An analysis of factors affecting price volatility of the US oil market. Energy Economics 24(2), 107–119 (2002)
2. Zhang, X., Lai, K.K., Wang, S.Y.: A new Approach for Crude Oil Price Analysis Based on Empirical Mode Decomposition. Energy Economics 30(3), 905–918 (2008)
3. Wang, S.Y., Yu, L., Lai, K.K.: Crude oil price forecasting with TEI@I methodology. Journal of Systems Sciences and Complexity 18(2), 145–166 (2005)
4. Yu, L., Wang, S.Y., Lai, K.K.: Forecasting Foreign Exchange Rates and International Crude Oil Price Volatility — TEI@I Methodology. Hunan University Press, Changsha (2007)
5. Yu, L., Wang, S.Y., Lai, K.K.: A Rough-Set-Refined Text Mining Approach for Crude Oil Market Tendency Forecasting. International Journal of Knowledge and Systems Sciences 2(1), 33–46 (2005)
6. Fan, Y., Liang, Q., Wei, Y.M.: A Generalized Pattern Matching Approach for Multi-Step Prediction of Crude Oil Price. Energy Economics 30(3), 889–904 (2008)
7. Box, G.E.P., Tiao, G.C.: Intervention Analysis with Applications to Economic and Environmental Problems. Journal of the American Statistical Association 70, 70–79 (1975)
8. Bonham, C.S., Gangnes, B.: Intervention Analysis with Cointegrated Time Series: The Case of the Hawaii Hotel Room Tax. Applied Economics 28(10), 1281–1293 (1996)
9. Mackinlay, A.C.: Event Studies in Economics and Finance. Journal of Economic Literature 35, 13–39 (1997)
10. Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Zheng, Q., Yen, N.-C., Tung, C.C., Liu, H.H.: The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis. Proceedings of Royal Society of London 454, 903–995 (1998)
11. Rilling, G., Flandrin, P., Goncalves, P.: On Empirical Mode Decomposition and Its Applications. In: IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing NSIP-03. Grado(I) (2003)

12. Wu, Z., Huang, N.E.: Ensemble Empirical Mode Decomposition: A Noise-Assisted Data Analysis Method. Centre for Ocean-Land-Atmosphere Studies, Technical Report No. 193, 51 (2004a), http://www.iges.org/pubs/tech.html
13. Wu, Z., Huang, N.E.: A study of the characteristics of white noise using the empirical mode decomposition method. Proceeding of Royal Society of London 460, 1597–1611 (2004)
14. Cecchetti, S.: Monetary Policy and the Financial Crisis of 2007-2008. CEPR Policy Insight (2008)
15. Wu, Z., Huang, N.E., Long, S.R., Peng, C.K.: On the Trend, Detrending, and Variability of Nonlinear and Nonstationary Time Series. Proceedings of the National Academy of Sciences 104, 1488–1489 (2007)
16. Ferderer, J.P.: Oil price volatility and the macroeconomy. Journal of Macroeconomy 18, 1–26 (1996)

Preface for the Joint Workshop on Tools for Program Development and Analysis in Computational Science and Software Engineering for Large-Scale Computing

Andreas Knüpfer[1], Arndt Bode[2], Dieter Kranzlmüller[3], Daniel Rodrìguez[4], Roberto Ruiz[5], Jie Tao[6], Roland Wismüller[7], and Jens Volkert[8]

[1] Center for Information Services and High Performance Computing
Technische Universität Dresden, Germany
[2] Lehrstuhl für Rechnertechnik und Rechnerorganisation
Technische Universität München, Germany
[3] Ludwig-Maximilians-Universität München (LMU) &
Leibniz Supercomputing Centre (LRZ), Munich, Germany
[4] University of Alcalá, Spain
[5] Pablo de Olavide University of Seville, Spain
[6] Steinbuch Center for Computing
Karlsruhe Institute of Technology, Germany
[7] Operating Systems and Distributed Systems
University of Siegen, Germany
[8] Institute of Graphics and Parallel Processing
Johannes Kepler University Linz, Austria

Today, computers and computational methods are increasingly important and powerful tools for science and engineering. Yet, using them effectively and efficiently requires both, expert knowledge of the respective application domain as well as solid experience applying the technologies. Only the combination allows new and faster advancement in the area of application. The same is true for establishing new computational concepts as regular methods in the field of application. This applies to either quantitative improvement (e.g. by parallel scalability) or by qualitative progress (e.g. by better algorithms).

Of course, scientists and engineers are most interested in solving the actual task. At the same time, computational tools require *some* knowledge about their usage and its implications. Yet, the tools must not demand intimate skills using the tool nor specialized computer science knowledge. Otherwise, the costs of tool usage and training will outweigh the benefits and it will not attract a broad user community.

The same applies in the area software engineering. The research about software engineering is being influenced by computational applications and vice versa. In one direction, software engineering methods, processes, metrics, management, etc. need to consider the way these types of applications are developed and executed. In the other direction, computational techniques can help to improve the accuracy and control of all types of projects.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 655–656, 2009.

Our workshop addresses tools and methods provided *by* computer scientists *for* scientists and engineers from their respective application domains. This includes the following:

- Software development tools
- Testing and debugging tools
- Program analysis and visualization tools
- Performance analysis and tuning tools
- Management of large amounts of data and data mining
- Software development processes
- Computational intelligence techniques applied to software engineering
- Data mining software engineering repositories
- Resource management, load balancing, job queuing and accounting
- Problem solving environments for specific application domains
- Use cases and practical experiences with real-world applications

Furthermore, it covers reports about use cases and success stories using the computational tools for science and engineering by either the users or by the computer scientists or by collaboration of both.

The primary intention of this workshop is to bring together developers of tools for scientific computing and their potential users. Since its beginning at the first ICCS in 2001, the workshop has encouraged tool developers and users from the scientific and engineering community to exchange their experiences. Tool developers present to users how their tools support scientists and engineers during program development and analysis. Tool users report their experiences employing such tools, especially highlighting the benefits as well as the desired improvements.

Snapshot-Based Data Backup Scheme: Open ROW Snapshot*

Jinsun Suk, Moonkyung Kim, Hyun Chul Eom, and Jaechun No

Dept. of Computer Software
College of Electronics and Information Engineering
Sejong University, Seoul, Korea

Abstract. In this paper, we present the design and implementation details of the Open ROW Snapshot which is the data backup scheme based on the snapshot approach. As the data to be stored in storage systems are tremendously increased, data protection techniques have become more important to provide data availability and reliability. Snapshot is one of such data protection techniques and has been adopted to many file systems. However, in large-scale storage systems, adopting a snapshot technique to prevent data loss from intentional/accidental intrusion is not an easy task because the data size being backup-ed at a given time interval may be huge. In this paper, we present the Open ROW Snapshot that has been implemented based on the file system-based snapshot approach. The Open ROW Snapshot provides a widely portable structure and causes less I/O processing overhead than the ROW(Redirect-On Write) method does. Furthermore, the Open ROW Snapshot provides a capability of maintaining the disk space assigned to snapshot images in a consistently-sized disk portion. We present the performance results of the Open ROW Snapshot obtained from the Linux cluster located at Sejong University.

1 Introduction

Many data recovery approaches [11,12,13] have been developed to protect important data against system crash. Especially, the snapshot-based data recovery has been adopted to many file systems [5,6,7,8,9,10] to provide data availability and reliability. However, in large-scale storage systems, implementing a snapshot technique to prevent data loss from intentional/accidental intrusion is not an easy task because the data size being backup-ed at a given time interval may be huge. Simply duplicating the inodes and data blocks associated with a point-in-time snapshot causes high I/O processing overhead. Furthermore, maintaining a large number of snapshot images consume a large portion of disk space.

Snapshots can be built in two different ways; one is a volume-based snapshot in which the snapshot images are taken under LVM (Logical Volume Manager), and the other is a file system-based snapshot in which all the snapshot related operations are performed under the file system control. Even though the volume-based

* This work was supported by a Seoul R&BD program.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 657–666, 2009.

snapshot can efficiently manage snapshot images and disk space, it requires to preserve some disk space for retaining snapshot images. The file system-based snapshot does not need to reserve the disk space to store snapshot images. However, because the file system-based snapshot is tightly coupled to the underlying file system, porting a snapshot implementation among several file systems would take considerable overheads. Porting a snapshot becomes even worse when file systems that are supposed to use a snapshot support different block allocation policies.

We developed the Open ROW Snapshot that combines the good features of both file system-based method and ROW approach. Our primary objectives in developing the Open ROW Snapshot were to minimize I/O processing overhead occurred between successive snapshot images, to provide a wide range of portability by supporting both the extent-unit and block-unit allocation policies, and to provide a capability of managing disk space for snapshots in a consistent size of disk section. In order to minimize I/O processing overhead occurred in duplicating inodes and data blocks to take a point-in-time snapshot, we chose to adopt the ROW-based snapshot approach. Besides, we used a pre-allocated metatdata to reduce the block allocation time for each instantaneous snapshot. Furthermore, the Open ROW Snapshot can easily be combined with the extent-based storage structure [1,2], as well as be combined with the block-based storage structure [3,4]. When the Open ROW Snapshot is combined with an extent-based storage structure, it can easily detect the sharing of data blocks between several snapshot images by checking the value of the bitmap flag. This helps to efficiently eliminate the corrupted snapshot images in the snapshot history.

The rest of this paper is organized as follows. In Section 2, we discuss the design motivations of the Open ROW Snapshot. Section 3 describes the implementation details of the Open ROW Snapshot and in Section 4, we present the performance results obtained from the Linux cluster at Sejong University. In Section 5, we conclude our paper.

2 Design Motivation

2.1 Minimize I/O Processing Time

In order to minimize I/O processing overhead occurred in duplicating inode and data blocks to take a point-in-time snapshot, we chose to adopt the ROW-based snapshot approach. Additionally, in adopting the ROW approach, we used a pre-allocated metadata to reduce the block allocation time for each instantaneous snapshot. When the inode of an active file is created, we also allocate an additional inode to be used for the following snapshot image. When a point-in-time snapshot is taken, this additional inode becomes an inode of the snapshot file. Also, instead of simply duplicating all the associated blocks of the snapshot file, the snapshot inode just links pointers to the original blocks to denote that these data blocks are shared between the active file and the snapshot file. In this way, we can significantly reduce the processing time for the block allocation, compared to that of COW(Copy On Write) approach.

2.2 Provide a Wide Range of Portability

The Open ROW Snapshot can easily be combined with the extent-based storage structure where a contiguous number of blocks are allocated to a file segment, as well as can be combined with the block-based storage structure. When the Open ROW Snapshot is combined with the extent-based structure, such as XFS [1,2], each inode, including the inode of a snapshot image, contains the extents composed of three components; the starting block address, block count describing the number of blocks contiguously allocated, and bitmap flag describing the sharing of the data blocks belonging to the extent. If the bitmap flag is set to 1, it would then denote that the data blocks belonging to the corresponding extent are recently allocated, and thus no other inode currently shares these data blocks yet. If the flag is set to 0, it would then mean that the data blocks belonging to the associated extent are shared between files, thus a careful block management is required while the modification to the data blocks happens to these blocks.

2.3 Manage Disk Space for Snapshots

We developed a snapshot spatial algorithm that enables us to keep the disk space allocated for snapshot images as smallest as possible. In the Open ROW Snapshot, all the snapshot images, including their active file, are grouped by two pointers, `prev_core_snap` and `next_core_snap`, and in each snapshot group, the active file is linked at the front and the oldest snapshot image is linked at the back. Constructing a snapshot group of an active file enables us to easily trace back and forth the link to find which data blocks are shared between files. When the Open ROW Snapshot finds corrupted or backup-ed snapshot images while traversing a snapshot group, the snapshot spatial algorithm can easily check the sharing of the data blocks belonging to those images. It also unlinks the associated inode from the snapshot group, and thus enables to keep snapshot images in a consistently-sized disk section.

3 Implementation Details

3.1 Overall Structure

Figure 1 illustrates an overview of the Open ROW Snapshot. In this Figure, the Open ROW Snapshot was configured to support for the extent-based allocation. For two active files `File1` and `File2`, there exist two snapshot files each, `snapshot1` and `snapshot2`. The Open ROW Snapshot uses two pointers, `prev_core_snap` and `next_core_snap`, to easily make a snapshot group of an active file, and thus to verify the integrity of each snapshot file with low processing overhead. The file pointed by `prev_core_snap` is of a preceding snapshot image, while conversely, the file pointed by `next_core_snap` is of a following snapshot image or an active file.

Figure 1 shows how the snapshot images are grouped as time goes by. At *t0*, two active files, `File1` and `File2` are created, and then, at *t1*, these two

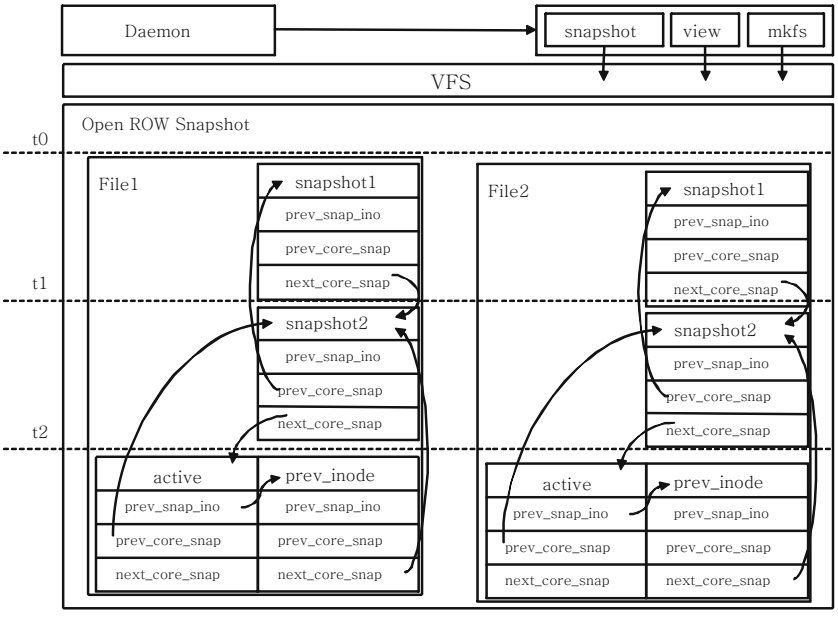


Fig. 1. An overview of the snapshot structure

files became the snapshot images of their active files, while being linked with their active files by using the `prev_core_snap`. To easily find out which is the next snapshot image of a file, each snapshot image is linked with the following one by using the `next_core_snap`. Grouping the related snapshot images with these two pointers enables for the Open ROW Snapshot efficiently to perform the snapshot spatial algorithm to eliminate a corrupted snapshot image or a backup-ed snapshot image. As a result, the snapshot files can be stored in a small-sized disk space.

When the inode of an active file is created, an additional inode, linked with the active inode by the `prev_snap_ino`, is also pre-allocated to be used as the inode of the following snapshot image. When the next snapshot is taken, there is no need to allocate and to replicate the inode of its active file. Only thing to be performed at that time is to adjust two pointers, `prev_core_snap` and `next_core_snap`, and to setup the bitmap value of the extent to denote the sharing of the data blocks. The Open ROW Snapshot provides a snapshot daemon that periodically wakes up and issues a `snapshot` system call to check the state of each snapshot image.

3.2 Snapshot Procedure

The Open ROW Snapshot assigns a bitmap value to each extent structure to manage the sharing of the data blocks, as shown in Fig. 2(a). If the bitmap value is set to `0`, it then means that the data blocks of the extent can not be modified because those blocks must have been shared with other files. Otherwise, the

blocks of the extent can be modified. Figure 2(a) shows an active file using the extent storage structure. The file is composed of five data blocks, `B0` through `B4`, and its extent includes three components: the starting block number, block count and bitmap value. Because there is no snapshot taken yet, the bitmap value is set to `1`, meaning that no other file currently shares the data blocks belonging to this file. Figure 2(b) describes the steps involved in taking the first snapshot. The file located at the left side in Fig. 2(b) denotes an active file and the file at the right side denotes its point-in-time snapshot image.

As can be seen in Fig. 2(b), the bitmap value of the active file is changed from `1` to `0` at the time of taking the first snapshot image because the data blocks of the active file are shared with those of the first snapshot. Changing a bitmap value has a significant performance impact on data modification or deletion because, by checking the current value of the bitmap, we can determine if the corresponding data block enables to be updated or the new blocks must be allocated to get the new data values.

Figure 2(c) describes the steps involved in the first update occurred after the first snapshot image was taken. In Fig. 2(c), the update requires two blocks, `B3` and `B4`, to be modified. Since these two blocks are shared between the active file and the first snapshot image, two new blocks, `B5` and `B6`, are allocated and then the update is performed on these two new blocks, without touching the two original blocks, `B3` and `B4`. Also, to reflect the new block allocation for the update, an additional extent is created and its bitmap value is initially set to `1` because no other file currently shares these new data blocks with the original file.

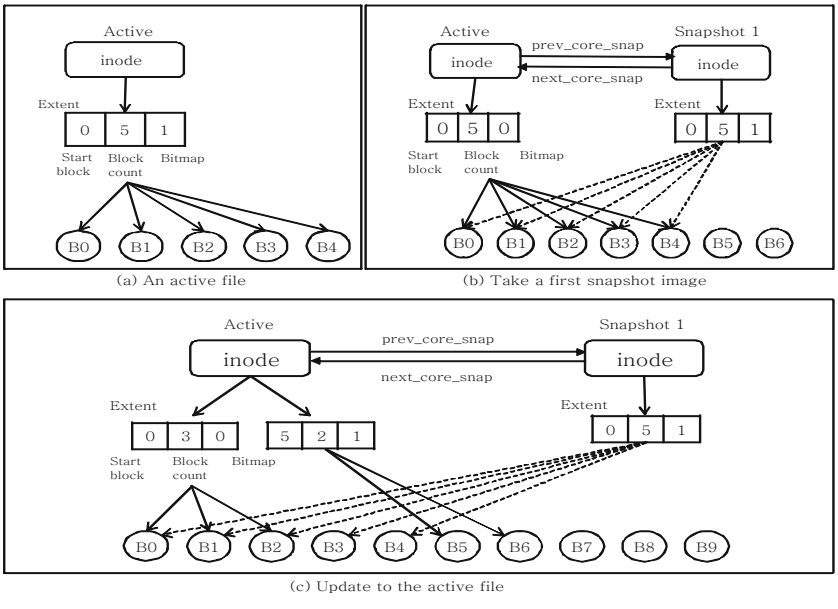


Fig. 2. Snapshot procedure in Open ROW Snapshot

3.3 Spatial Optimizations

In order to minimize I/O processing overhead to be occurred while a snapshot is taken, the Open ROW Snapshot pre-allocates the inode for the next snapshot and duplicates all the data block addresses to the pre-allocated inode. Furthermore, to maintain the snapshot image groups in a small-sized disk section, the snapshot daemon is periodically waken up to make sure that all the snapshot images are in a consistent state. When the snapshot daemon finds a snapshot image that has been corrupted or an image that has been backup-ed to other disk, the daemon eliminates the snapshot by unlinking it from the snapshot history.

The Open ROW Snapshot provides a snapshot spatial algorithm in which any snapshot image linked at the middle of the history can efficiently be deleted by checking the sharing of the data blocks. Figures 3(a) and (b) show the steps involved in the snapshot spatial algorithm to eliminate a snapshot image, `snapshot2`. Figure 3(a) describes a snapshot overview before eliminating a corrupted snapshot image, `snapshot2`. In this Figure, the first snapshot image, `snapshot1`, includes an extent denoting that four data blocks, `B0` through `B3`, are allocated to this image. The second snapshot image, `snapshot2`, inherits four data blocks from `snapshot1`, while changing the bitmap value from `1` to `0` because those data blocks are shared between these two files. It is noted that when `snapshot2` was an active file, there existed a write operation requiring three new

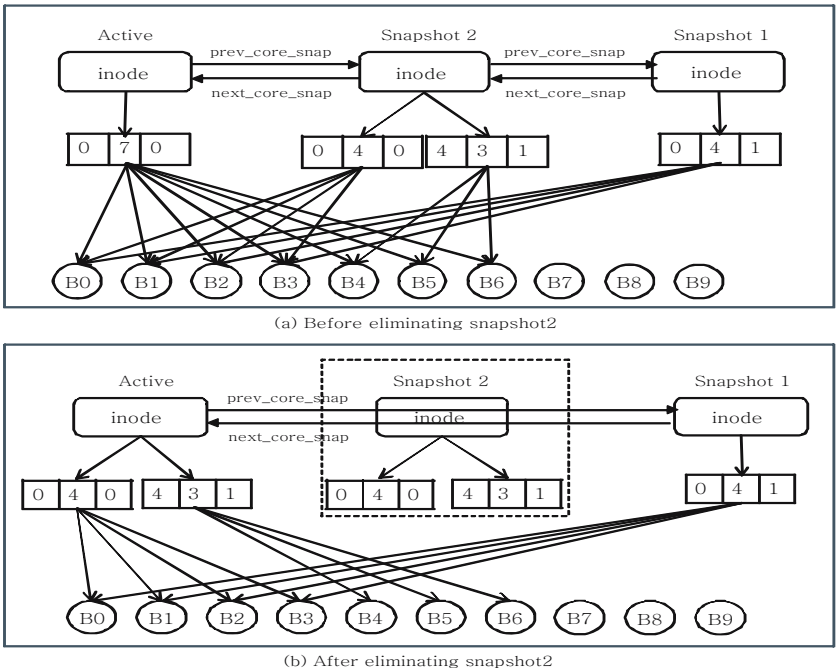


Fig. 3. Snapshot elimination process

data blocks, B4 through B6, to be allocated. Therefore, the Open ROW Snapshot assigns an extent to the inode of snapshot2 while setting its bitmap value to 1 because no sharing happened yet to those new blocks.

When the second snapshot was taken, the active file contains these seven data blocks, B0 through B6. Suppose that the snapshot daemon finds the second snapshot image, snapshot2, was corrupted. The elimination process occurred in the snapshot spatial algorithm requires the daemon to traverse prev_core_snap and next_core_snap pointers to determine if the data blocks belonging to the file to be deleted can be deallocated. The snapshot spatial algorithm to eliminate a corrupted image works as follows.

- In case that the bitmap value of an extent is of 0. It denotes that the blocks belonging to this extent can not be modified because those blocks are shared with other file linked by the prev_core_snap pointer.
- In case that the bitmap value of an extent is of 1. It denotes that the blocks belonging to this extent is not shared with the preceding snapshot images. However, these data blocks can be shared with other following snapshot images connected to by the next_core_snap pointer. Therefore, the traversal through the next_core_snap pointer is needed to execute.

In order to eliminate snapshot2, the bitmap value of all the extents of snapshot2 should be checked. As can be seen in Fig. 3(b), snapshot2 has two extents to manage the data blocks associated. Since the bitmap value of the first extent is set to 0, the corresponding data blocks can not be deallocated, and thus the snapshot spatial algorithm simply unlinks the pointers to those data blocks. The bitmap value of the second extent is of 1, therefore the preceding snapshot images connected to by the prev_core_snap pointer with snapshot2 has not shared the data blocks with snapshot2. However, since these data blocks can be shared with the following snapshot images or the active file, before de-allocating the data blocks, the spatial algorithm should take into account the bitmap value of the active file labeled as active in Fig. 3(b). Because the active file includes the data blocks being shared with snapshot2, the snapshot spatial algorithm splits the extent into two parts to separate the data blocks. The bitmap value of the first split extent is set to 0 because, even though snapshot2 is eliminated, the associated data blocks, B0 through B3, are still shared with the first snapshot image, snapshot1. On the other hand, the bitmap value of the second split extent is set to 1, because, after eliminating snapshot2 image, no other file shares the associated data blocks, B4 through B6, with the active file.

4 Performance Evaluation

We obtained all performance results on the Linux cluster at Sejong university. We installed the Open ROW Snapshot on top of XFS using the extent-based allocation policy and produced the performance results. Figures 4 through 9 compare the performance measurements of the Open ROW Snapshot structure and the LVM snapshot. Figures 4 and 5 show the results measured using small

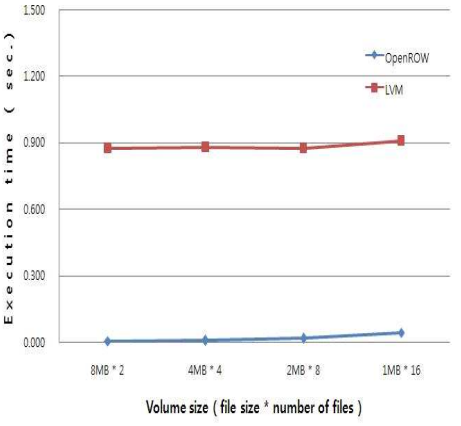


Fig. 4. Snapshot execution time using the small size of files (1MB - 8MB)

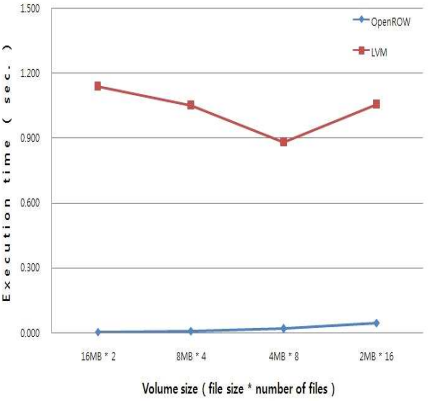


Fig. 5. Snapshot execution time using the small size of files (2MB - 16MB)

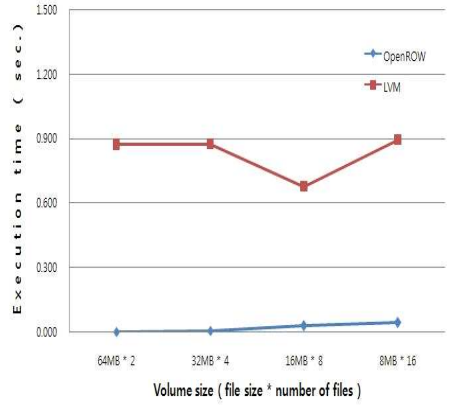


Fig. 6. Snapshot execution time using the intermediate size of files (8MB - 64MB)

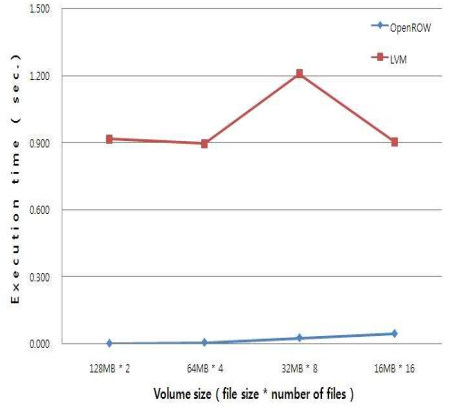


Fig. 7. Snapshot execution time using the intermediate size of files (16MB - 128MB)

files. At each experiment, we calculated the volume size to be the file size multiplied by the number of files. With the LVM snapshot, no matter which files need to be taken pictures, the total volume size should be duplicated, therefore, the time for taking a snapshot image is quite high. On the other hand, the Open ROW Snapshot only takes duplicated images for the files that are not either backup-ed, or corrupted. Furthermore, in order to minimize the I/O processing overhead, we used the pre-allocated inode for a following snapshot image. These optimizations help to produce better performance in the Open ROW Snapshot than in the LVM snapshot. Quite similar results are obtained with the

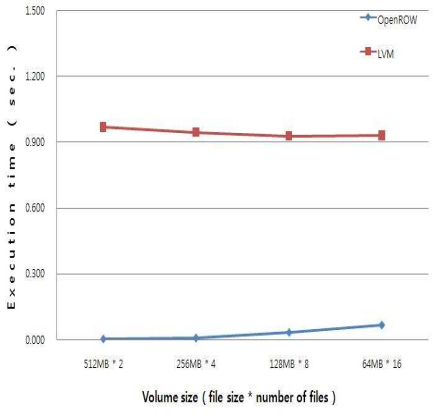


Fig. 8. Snapshot execution time using the large size of files (64MB - 512MB)

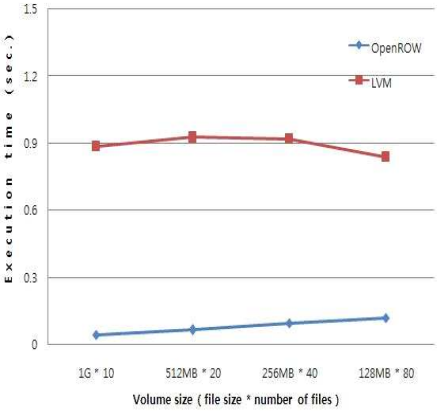


Fig. 9. Snapshot execution time using the large size of files (128MB - 1GB)

intermediate file size, as shown in Fig. 6 and Fig. 7, and with the large file size, as shown in Fig. 8 and Fig. 9.

5 Conclusion

In this paper, we presented the Open ROW Snapshot that combined the good features of both file system-based method and ROW approach. The Open ROW Snapshot was designed to minimize I/O processing overhead occurred between successive snapshot images, to provide a wide range of portability by supporting both the extent-unit and block-unit allocation policies, and to provide a capability of managing disk space for snapshots in a consistent size of disk section. We presented the performance results obtained from the Linux cluster at Sejong university. With the LVM snapshot, no matter which files need to be taken pictures, the total volume size should be duplicated, therefore, the time for taking a snapshot image is quite high. On the other hand, the Open ROW Snapshot only takes duplicated images for the files that are not either backup-ed, or corrupted. This optimization helps to produce better performance in the Open ROW Snapshot than in the LVM snapshot. In the future work, we will justify and improve the performance of the Open ROW Snapshot by doing more experiments.

References

1. Sweeney, A., Doucette, D., Hu, W., Anderson, C., Nishimoto, M., Peck, G.: Scalability in the XFS File system. In: USENIX 1996: Annual Technical Conference (1996)
2. Mostek, J., Earl, W., Koren, D.: Porting the SGI XFS File System. In: Linux 6th Linux Kongress: The Linux Storage Management Workshop, LSMW (1999)

3. Peterson, Z.N.J., Burns, R.C.: Ext3cow: The design, implementation, and analysis of metadata for a timeshifting file system. Technical report, Department of Computer Science, The Johns Hopkins University (2003)
4. Peterson, Z.N.J., Burns, R.C.: Ext3cow: A Time-Shifting File System for Regulatory Compliance. ACM Transactions on Storage 1(2), 190–212 (2005)
5. Shim, S., Lee, W., Park, C.: An Efficient Snapshot Technique for Ext3 File System in Linux 2.6. Technical report, Pohang University of Science And Technology
6. Americal Megatrends. Inc., AMI Snapshot Thechnology, Technical report (2005)
7. Chapman, D., Merrill, J.: Open systems SnapVault. Technical Report 3466, Network Appliance (2006)
8. Mary, B., Perterson, D.: Integrating Network Appliance Snapshot and SnapRestore with Veritas Netbackup in an Oracle Backup Environment. Technical Report 3394, Network Appliance (2006)
9. Patterson, H., Manley, S., Federwisch, M., Hitz, D., Kleiman, S., Owara, S.: SnapMirror: File System Based Asynchronous Mirroring for Disaster Recovery. In: Proceedings of the FAST 2002 Conference on File and Storage Technologies, pp. 28–30 (2002)
10. SuSE Inc., The Logical Volume Manager (LVM), Technical report (2002)
11. Piernas, J., Cortes, T., Garcia, J.: DualFS: a New Journaling File System without Meta -Data Duplication. In: Proceedings of the 2002 International Conference on Supercomputing (2002)
12. Seltzer, M., Bostic, K., McKusick, M.K., Staelin, C.: An Implementation of a Log-Structured File System for UNIX. In: USENIX Annual Technical Conference (1993)
13. Santry, D., Feeley, M., Hutchinson, N., Veitch, A.: Elephant: The File System that Never Forgets. In: Proceedings of IEEE Hot Topics in Operating Systems (1999)

Managing Provenance in iRODS

Andrea Weise[1], Adil Hasan[2], Mark Hedges[3], and Jens Jensen[4]

[1] Centre for Advanced Computing and Emerging Technologies (ACET), University of Reading, UK
a.weise@reading.ac.uk
[2] English Department, Liverpool University, UK
adilhasan2@googlemail.com
[3] Centre for e-Research, King's College London, UK
mark.hedges@kcl.ac.uk
[4] Science and Technology Facilities Council, Rutherford Appleton Laboratory, UK
jens.jensen@stfc.ac.uk

Abstract. Nowadays provenance is an important issue. Provenance data does not only give a history of events, it also provides enough information to allow the opportunity to verify the authenticity of the data, as well as, determine the quality of the data. The data grid management system, iRODS, comes with metadata which can be used as provenance data. Currently, iRODS's metadata is not sufficient for tracking and reconstructing procedures applied to data. In this paper, we describe the provenance needs of iRODS and we survey briefly current provenance and provenance enabled workflow systems. We describe an architecture that can be used to manage provenance in iRODS (and other systems) in a fault-tolerant way.

1 Introduction

1.1 Background

The work presented in this paper grew out of investigations in the "Architecture for a Shibboleth-Protected iRODS System" ASPiS project. The aim of this project is to add support for provenance capture to the Rule Oriented Data System (iRODS), to develop Shibboleth single sign-on access to iRODS, and to deploy this for groups of users who are currently using the Storage Resource Broker (SRB). iRODS is developed by the Data Intensive Cyber Environments (DICE) group (developers of the SRB), and collaborators [1].

This paper presents two results, first an evaluation of two existing general purpose provenance systems with respect to use cases provided by our current users of SRB. Secondly, based on this evaluation we describe work done in the ASPiS project to develop a generic provenance system for a distributed environment, in particular for the data grid management system iRODS.

1.2 iRODS

The Rule Oriented Data System (iRODS) [1], is an implementation of a "data grid". It is often seen as the successor to the Storage Resource Broker (SRB).

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 667–676, 2009.

Both systems are able to provide uniform access to heterogeneous storage devices across the network and, as a result, make the storage infrastructure appear transparent to the end user [2]. The significant difference between the SRB and iRODS lies in the way the data can be managed within the system itself. iRODS, recently released as version 2.0, comes with a rule engine. With the introduction of rules, iRODS is able to adapt to many scenarios and the user is able the manage their own data in almost any way. Additionally, the implementation of services to manage or process the data, such as data conversion, can be easily achieved by the end user. The rule engine enforces rules and therefore, acts as interpreter of the rules [3]. Rules are composed of the actual event, conditions, action sets and recovery sets [4], and applied through microservices. Microservices are functions written in C [1], which can be provided by anyone to organise and structure the data. Through rules and microservices different applications can be accessed by iRODS, e.g. communication to a web service or data conversion, and workflows can be defined. The rule engine is invoked from certain procedures within the iRODS code, at points which correspond to certain actions being performed; for example, just before (or just after) reading (or depositing) a file. The rule engine will first analyse the request to determine whether there is a rule defined in the rule base that corresponds to the action. If there is, the rule engine will execute the rule as defined (e.g. converting a deposited file to a different file format). If the rule executes successfully, the main processing continues from just after the point at which the engine was invoked. To manage the data, iRODS keeps "standard" metadata such as information about the file itself (e.g. file size, last accessed, owner, etc.) as well as user-defined metadata. The user-defined metadata can be connected with individual files which have been submitted to iRODS. The metadata are stored in the "Meta Data Base" (iCAT).

1.3 Related Work

In this project, provenance is seen as the history of the operations applied to a digital object. Other words for provenance can be history, pedigree, parentage, genealogy, filiation, or lineage[5]. The term "provenance" in scientific research can be defined in a number of ways in different contexts, [6], [7]. Based on the variety of different provenance systems, it can be said that provenance capturing systems play more and more of a dominant role in research. The Taverna [8] software from the myGrid project was primarily developed to manage user-defined workflows, and has a bioinformatic background. Some tools, such as VisTrails [6] and Taverna, aim to support the researcher by providing a platform that can combine different components, e.g different web services, and make them accessible through a single interface. Systems like REDUX [9] and the Virtual Data System (VDS) [10] use proprietary approaches to provide new ways of capturing and querying provenance data. Each of these systems were designed for a particular purpose and cannot easily be applied to any other existing system. Generic systems are provided by the Provenance Aware Service-Oriented Architecture (PASOA) and the Karma framework which we evaluate in this paper.

Another generic information system is Relational Grid Monitoring Architecture (R-GMA) from the "Enabling Grids for E-science in Europe" (EGEE) project [11]. Building on a producer/ consumer model, this system carries generic logging and accounting information in grids, and could be adapted to manage provenance data in a distributed and heterogeneous environment.

2 Requirements

Our use cases posed the following requirements:

1. Manage data throughout its lifecycle: from inception to final publication. It is the norm that data is written once and must not then be modified; when data is analysed, new files are created. Some data is not useful forever, or cannot be kept forever for other reasons, and may need to cleaned up.
2. Capture and record information about the data analysis: which files were analysed, how they were processed, where the result is stored.
3. Enforce proper ownership of data throughout its lifetime: for some of our users, the owner is the Virtual Organisation (VO). For others the owner is the *research proposal* which was submitted by the principal investigator – data created in the project is associated with the proposal throughout and the investigator can add or remove people. Ownership is used by investigators to define access control policies.
4. Ensure data access is auditable.
5. Ensure the infrastructure is *robust* and *scalable*.

We focus on the provenance related requirements, integrating them with iRODS-based data storage with callouts to systems managing provenance data, thus fulfilling most of the requirements.

All of our users have requirements for data provenance. Some of this data is (or can be) maintained natively by iRODS, such as file length, basic checksums, time created, etc. Checksums and file length are especially valuable provenance data because those information can be used to identify a particular file version, detect changes and verify integrity. For the remaining some can be automatically generated by custom built microservices, others must be provided by the user. We primarily focused on the former, and chose to keep the data in an external provenance system instead of storing it in the iCAT, because iRODS' built-in user metadata system is not built for provenance. Finally, the iRODS schema may change with new releases.

3 Evaluation of Karma and PASOA

In this section, we provide brief description of the two generic provenance systems we surveyed and evaluated.

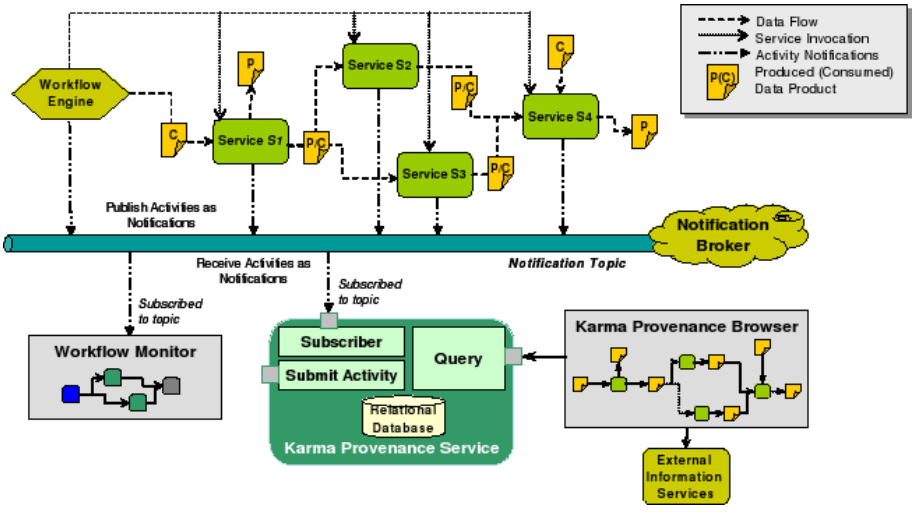


Fig. 1. Karma Provenance Framework

3.1 Karma

Karma aims to develop a provenance framework that will "record uniform and usable provenance metadata independent of the workflow or service framework used" [12]. The framework is written in Java and is therefore platform independent. Karma is used in an environment with workflows, services, service clients, and data products. The output of one service can serve as the input to the next [12]. Provenance data ("activities") within Karma are formatted XML messages which are submitted via a web services interface (synchronously). The Karma framework also provides an asynchronous publish-subscribe notification protocol based on [13]. Workflow engines and participating services publish their activities (events) to the notification broker. The subscriber and the Karma provenance service, respectively, subscribe to certain channels of interest and will get the information only for those channels they subscribed to. The publisher and subscriber are thus decoupled from each other and the subscriber does not necessarily know the actual source of the information [13]. Fig. 1 taken from [12] shows the interaction of the Karma2 framework with a workflow engine.

3.2 PASOA

An outcome of the "EU Provenance Project" funded by the European Commission's Sixth Framework Programme was the open provenance architecture [14]. The "Provenance Aware Service-oriented Architecture" – PASOA – project is based on that architecture.

PASOA aims to provide an independent and standardised protocol for capturing, recording, and accessing provenance which should be applicable to any

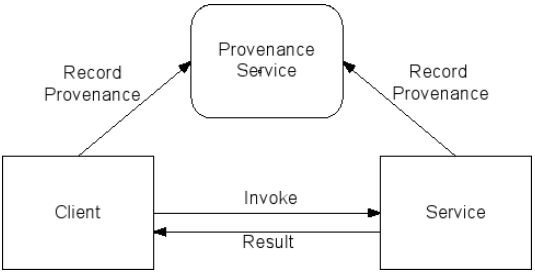


Fig. 2. The interaction between a client service and provenance service

system. The requirements in the project are trust, preservation, security, scalability, generality, and customisability [15]. These fit well with our requirements.

PASOA relies on a third party provenance service. One advantage of outsourcing the provenance service, according to PASOA, is that the workflow system itself does not have to deal with the provenance handling. Fig. 2 taken from [15] displays the basic approach of PASOA.

Each participant of a collaboration sends information to the provenance service. The provenance service itself is a web service with a local storage mechanism, e.g. a database. PASOA implements its own proprietary communication protocol. Future plans for PASOA are to provide eventually a protocol, which can serve as a standard for provenance capturing. Currently, the protocol consists of four phases: negotiation, invocation, provenance recording, and termination. The negotiation phase is used to arrange which of the provenance services is used, e.g. recording or querying. The invocation phase will invoke the actual service which was determined during the previous phase. Within the provenance recording phase, the provenance data is submitted to PASOA, whose success will be confirmed by sending a finished message in the termination phase[15].

4 Architecture

4.1 Using iRODS to Manage Provenance

Extending previous work in [6], we distinguish between capturing, recording and storing, processing, displaying, and querying provenance metadata. Capturing needs to recognise the data that is coming in. For recording and storage, we develop a way for microservices to call out to a provenance store (web service). We are not currently addressing display of provenance metadata within the ASPiS project. Until such features can be implemented, there are external applications that let users view and query the metadata held in the provenance store.

iRODS does not by default capture changes made to data in rule-based workflows. We can not rely on the user entering it – the user may be absent or may not know the workflow in detail. The metadata kept by iRODS itself is not sufficient: it does not capture the workflow.

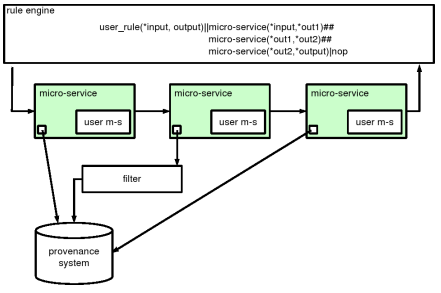


Fig. 3. Microservice Wrapper

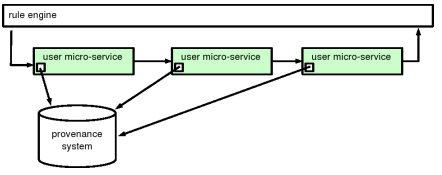


Fig. 4. User Microservice Chain

We looked at two ways of capturing metadata about the workflow within iRODS. Figure 3 shows a wrapper microservice which captures all information (but may in principle filter it before recording it). Figure 4 shows user microservices modified to record provenance metadata. The former has the advantage of requiring no changes to the user microservices; all metadata is recorded. Another advantage is that it can be used to optimise the workflow because it can keep track of profiling data for the microservices. The disadvantage is that, in general, it does not know about the data and is not able to capture specific user defined information. In this project we chose to focus on the latter case, where user microservices are modified to record provenance data. This enables us to record precisely the data that is needed, which may include the following[1]:

- User defined provenance data.
- Data about the user: identity, authorisation such as roles and group memberships, home institution, potentially other Shibboleth attributes (subject to data protection and other rules)
- File provenance data: filename, length, checksum, dates written and modified, owner, and various types of integrity management metadata: checksums, number of copies held in the underlying storage, where they are physically stored, their access latency. Which other iRODS services have this data?
- Access to data: who accessed it and did what to it, why they were granted permission?
- Content provenance data: what is contained in the file, what is the format of the file, version of file (a version that makes sense to the user and one that makes sense to iRODS).
- Which microservices, and which version of the services, have processed the data.
- Which rules and which version of rules, have processed the data.
- We discussed keeping track of versions and configurations of iRODS itself. Changing the configuration or upgrading iRODS to a new version will not change the data, but might introduce changes to the iCAT. Moreover, our use cases may require users using microservices in iRODS to manage their

[1] Some of this is relevant to provenance and some of it is data storage "housekeeping".

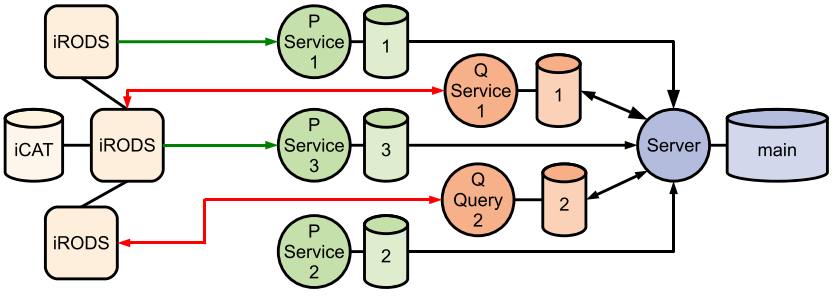


Fig. 5. iRODS Provenance Framework Overview

analysis workflow, and an upgrade to the system could change the way these are run.

Since iRODS is able to contact existing applications such as external workflow systems there are issues with how to extract provenance data from an external system, how to relate it to the iRODS provenance information, and how that information is stored (format). However, this is currently out of the scope in this project.

4.2 Proposed Provenance System

PASOA and Karma rely on a web service to be provenance aware application independent. In a distributed environment with an unknown amount of nodes sending requests, a single web service can become a single point of failure. For both systems it can be said, if either the web service or the database behind the system fails, the entire provenance capturing framework will fail. Further, the single web service can also become a performance bottleneck.

To resolve the problem of having a single point of failure, the querying and recording services will have several access points. Fig. 5 shows the outline for the proposed distributed provenance system which will be used in connection with iRODS, whereas iRODS will be able to access the web services through microservices as described in chapter 4.1.

We distinguish between web services for querying (in the following referred to as "Q-Services") and those for recording provenance metadata ("P-Services"). Each service has its own local storage mechanism (e.g. database) which serves as storage cache and replicates part of the main database. This will reduce the traffic to the core data storage device as well as increase the response time for the Q-Services. Each service caches its storage content independently. There will be a constant synchronisation between the main database and each P-Service cache to ensure consistency with a configurable synchronisation interval. Consistency can be forced by using triggers. In that case, the content will be forwarded immediately to the main storage device. The Q-Services will try to answer queries from the data contained in the attached database. If the service can not answer the

query, it will forward the request successively to n other services where n is configurable. This forwarding will reduce the traffic to the core database and will in general reduce the response time for the requesting client. If a query service gets a forwarded query, it will respond to the sender with either a positive or negative acknowledgement. If the requested data is available (positive acknowledgement), the service will respond directly to the client by returning the query results. If a sending Q-Service receives a negative acknowledgement, the next service in the queue is contacted. If all attempts fail to find another suitable query service, the original contacted Q-Service will request the data from the core database. It can be assumed that the possibility for the client to contact the same service with a similar query is higher than for the client to contact a different service due to the way web services are discovered. Introducing a cache control system can reduce the time of finding another suitable query service and can control the number of requests for each service. But such a system is currently out of scope for this project. The need to have a cache control system e.g. proposed by Katoaka et. al [16] can be analysed after a successful implementation and resultant tests.

If a query is submitted before the data has been migrated to the central database, the Q-Service will fail to find the data as it cannot query P-Service directly. The use of triggers in P-Services and the core database service can reduce the delay for data which needs to be highly available. Although the main database should be backed up, the contents can be partly recovered by resynchronising with the P- or Q-Services databases.

When implemented in an iRODS-based data grid, each node will discover its nearest P-Service using the peer-to-peer (P2P) based approach of a "balanced distributed web service" look-up system described in [17]. Furthermore, each iRODS node will cache the address of its last accessed P or Q service.

5 Discussion

5.1 Technical Issues

Karma and PASOA provide web services for recording and querying metadata. We chose to use Tomcat to run the web services. The current Tomcat 6.x release, published under the Apache Software License, is the result of a nine year development and the acceptance in the user community is very high. Therefore, we consider this software product mature. In addition, our tests on PASOA showed that Tomcat itself was very stable.

Since microservices are written in C, and provenance services in Java, they are best linked using web services. To implement web services in C, we generate code with gSOAP, a cross-platform open source kit which is able to generate platform independent C/C++ source code based on a given WSDL file. We discovered that gSOAP was sensitive to the WSDL input, and a namespace compatibility problem had to be debugged. Moreover, some WSDL files generated code which could not be linked. Once these problems were fixed, however, we managed to get a stable and reliable interface.

5.2 Future Work

Within the ASPiS project we have looked at recording, storing, and querying provenance metadata. We have not looked in depth at displaying metadata, but there are tools that can be used to query data captured by PASOA [18]. In future work, we will look at closer integration with such tools, as well as workflow and other computational metadata, particularly for the National Grid Service and the portals used by the users of this service.

We will also need to look at the mechanism for registering microservices and maintaining the versions and integrity of them. Currently, we rely on the developers to maintain the version information, i.e., update it when the microservices is modified, but this does not protect against malicious modifications (or accidental ones, or developers who forget to update the version number).

6 Conclusion

In this paper we have enhanced iRODS by incorporating provenance functionality. As part of our research we surveyed existing provenance systems. The survey resulted in the fact that most provenance system or provenance enabled work flow systems are designed for a certain puprose and are therefore system dependent. Only the frameworks of PASOA and Karma offer system independency. iRODS rules and microservices were chosen to connect the data management system with such independent provenance frameworks. This way, there will be no interference with the iRODS core system and the emerging system can be seamlessly integrated in any distributed environment.

PASOA and Karma work with web services, which are the current state of the art of web applications. However, both provenance systems have a single point of failure which makes them difficult to use in a distributed environment. The proposed system eliminates this complication by dividing the provenance services into two major categories, submitting and querying services, which will enhance stability. By increasing the number of access points for those services and storage mechanisms, the new system becomes more fault tolerant and therefore, highly available. The access nodes will automatically scale by applying a P2P based algorithm which will provide an efficient and fault tolerant web service discovery mechanism.

Acknowledgement

This work, which is part of the ASPiS project, was funded by the UK Joint Information Systems Committee (JISC) as part of its e-Infrastructure programme, with additional support from UK Science and Technology Facilities Council (STFC). The authors would like to express their gratitude to the SRB and iRODS staff at SDSC and the University of North Carolina. The first author would like to express her gratitude to Prof. V. Alexandrov, ACET, Reading University.

References

1. About irods, https://www.irods.org/index.php/Introduction_to_iRODS
2. Rajasekar, A., Wan, M., Moore, R., Schroeder, W., Kremenek, G., Jagatheesan, A., Cowart, C., Zhu, B., Chen, S.Y., Olschanowsky, R.: Storage resource broker - managing distributed data in a grid. Technical report, San Diego Supercomputer Center (SDSC), University of California
3. Rule engine, https://www.irods.org/index.php/Rule_Engine
4. Rules, https://www.irods.org/index.php/Rules
5. Simmhan, Y.L., Plale, B., Gannon, D.: A survey of data provenance techniques. Technical report (2005)
6. Freire, J., Koop, D., Santos, E., Silva, C.T.: Provenance for computational tasks: A survey. Computing in Science & Engineering 10(3), 11–21 (2008)
7. Simmhan, Y.L., Plale, B., Gannon, D.: A framework for collecting provenance in data-centric scientific workflows. In: IEEE International Conference on Web Services, pp. 427–436 (2006)
8. Taverna 1.7.1 manual, http://www.mygrid.org.uk/usermanual1.7/
9. Barga, R.S., Digiampietri, L.A.: Automatic capture and efficient storage of e-science experiment provenance. Concurrency and Computation: Practice and Experience 20(5), 419–429 (2008)
10. Foster, I., Vöckler, J.S., Wilde, M., Zhao, Y.: Chimera: A virtual data system for representing, querying, and automating data derivation. In: SSDBM 2002: Proceedings of the 14th International Conference on Scientific and Statistical Database Management, Washington, DC, USA, pp. 37–46. IEEE Computer Society, Los Alamitos (2002)
11. R-gma: Relational grid monitoring architecture, http://www.r-gma.org/
12. Simmhan, Y.L., Plale, B., Gannon, D.: Karma2: Provenance management for data-driven workflows. Int. J. Web Service Res. 5(2), 1–22 (2008)
13. Eugster, P.T., Felber, P.A., Guerraoui, R., Kermarrec, A.M.: The many faces of publish/subscribe. ACM Computing Surveys 35, 114–131 (2003)
14. Moreau, L., Ibbotson, J.: The EU Provenance Project: Enabling and Supporting Provenance in Grids for Complex Problems (Final Report). Technical report, The EU Provenance Consortium (2006)
15. Groth, P., Luck, M., Moreau, L.: Formalising a protocol for recording provenance in grids. In: The UK OST e-Science second All Hands Meeting 2004, AHM 2004 (2004)
16. Kataoka, M., Toumura, K., Okita, H., Yamamoto, J., Suzuki, T.: Distributed cache system for large-scale networks. In: International Multi-Conference on Computing in the Global Information Technology, 2006. ICCGI 2006, p. 40 (August 2006)
17. Sioutas, S., Sakkopoulos, E., Drossos, L., Sirmakessis, S.: Balanced distributed web service lookup system. J. Netw. Comput. Appl. 31(2), 149–162 (2008)
18. The provenance architecture client side library, http://www.gridprovenance.org/software/CSLPage.html

Instruction Hints for Super Efficient Data Caches

Jie Tao[1], Dominic Hillenbrand[2], and Holger Marten[1]

[1] Steinbuch Center for Computing
Forschungszentrum Karlsruhe
Karlsruhe Institute of Technology, Germany
{jie.tao,holger.marten}@iwr.fzk.de
[2] Computer Laboratory
University of Cambridge, United Kingdom
dh378@cam.ac.uk

Abstract. Data cache is a commodity in modern microprocessor systems. It is a fact that the size of data caches keeps growing up, however, the increase in application size goes faster. As a result, it is usually not possible to store the complete working set in the cache memory.

This paper proposes an approach that allows the data access of some load/store instructions to bypass the cache memory. In this case, the cache space can be reserved for storing more frequently reused data. We implemented an analysis algorithm to identify the specific instructions and a simulator to model the novel cache architecture. The approach was verified using applications from *MediaBench*/*MiBench* benchmark suite and for all except one application we achieved huge gains in performance.

Keywords: Cache optimization, simulation, architecture design.

1 Introduction

The memory system is a traditional optimization target for enhancing the overall performance of a computational architecture. With the microprocessor design increasingly focusing on the power issue, memory also becomes the goal of optimization in terms of both performance and energy consumption.

To optimize the memory system, cache is the key. In the performance aspect, an access to the cache is much more efficient than a reference to the main memory because a cache access takes only several CPU cycles while an access to the main memory needs more than 100 cycles on modern processors. Hence, a lower cache miss rate can reduce the CPU time for executing an application. This actually also decreases the power requirement because accessing the main memory consumes more energy than accessing the cache.

The common approach for cache optimization focuses on improving the data locality so that data stored in the cache can be reused before being moved back to the main memory. This locality is typically achieved by restructuring the program code based on affinity analysis [8,10,12,13,14].

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 677–685, 2009.

We propose a different approach that enhances the cache efficiency, in terms of both performance and energy consumption, by storing only carefully selected data in the cache. We achieve this goal with an analysis algorithm that goes through the memory access trace of an application and thereby detects load/store instructions which do not introduce any cache hits. We decide not to load the data required by these instructions into the cache. In this case, cache space can be maintained for other data.

We then implemented a cache simulator for verifying the proposed approach and also for providing researchers a generic tool to study the impact of cache bypassing on the performance of applications. The simulator models the basic functions of a traditional cache with an additional functionality of bypassing the data of specific instructions.

The *MediaBench* and *MiBench* benchmarks for embedded system were used to examine the results of our cache optimization. As the cache is usually small on embedded systems it is more valuable to use their cache space efficiently. For the tested applications we achieved an average improvement of 2.7x in cache hit rate.

The remainder of the paper is organized as following. Section 2 first gives an overview of related work on cache optimization. This is followed by a detailed description of the proposed concept in Section 3. Section 4 presents the initial experimental results. The paper concludes in Section 5 with a short summary and several future directions.

2 Related Work

In the area of cache optimization, a lot of research work has been performed and different approaches were proposed. They either only address the performance aspect or also have specific mechanisms for power saving. Here, we focus on the latter.

Esakkimuthu et al. [5] deployed both hardware and software techniques to save the energy consumption of the memory system. The hardware approach includes a block buffering scheme and a sub-banking strategy. The former uses a buffer to store the previously accessed cache line. In this case, if the next data request targets on the same cache line, data can be acquired from the buffer without accessing the cache. In the sub-banking optimization, the data array of the cache is partitioned into banks and by a data request only the corresponding bank is accessed rather than the whole cache. The software approach applies the conventional code optimization schemes like loop interchange, loop tiling, and loop unrolling to improve the data reuse with a combined result of less cache misses and energy consumption.

Yang et al. [16] proposed a cache architecture to reduce the power demand but without performance degradation. This architecture deploys an L0, an instruction cache between CPU and the L1 instruction cache, to store frequently accessed instructions. A steering mechanism is employed to direct an instruction to the correct location. Simulation results show a 52% instruction cache energy reduction on average for a set of multimedia applications.

Benitez et al. [3] proposed a reconfigurable cache architecture that can be adapted to the running programs. The adaptation scheme is based on two techniques: a learning process provides the best cache configuration for each program phase, and a recognition process detects program phase changes. In addition, a low-overhead reconfiguration mechanism was designed. Simulation results show that this approach can achieve performance improvement and cache energy saving at the same time. Similarly, Cordon-Ross et al. [6] also address reconfigurable cache architectures but use heuristic tuning method to determine the best cache parameters. A specific contribution of this work is its mechanisms for tuning a data and instruction unified second level cache. As such tuning covers a large space, this approach has achieved a higher power saving than the other one. Abella et al. [1] also focus on the second level cache and designed a hardware technique to turn off those cache lines that are not expected to be reused. Alternatively, Ishihara et al. [9] use specific algorithm to place the code in the cache in a way that not all cache lines in a set are used. Unused cache lines can be disconnected for saving the power.

Overall, various schemes have been deployed in the last years for performance and energy issues of the memory system. They can be roughly classified to two groups: one focuses on improving the data locality with additional hardware support and the other on adapting the cache architecture to applications. Our approach can be classified to the first group because it also addresses on the locality issue. However, this approach is different from other work in that it allows some data accesses to bypass the cache memory so to save spaces for more important data.

3 The *Ihint* Architectural Approach

Key to our performance improvements is a novel cache architecture capable of bypassing the data of some loads/stores according to the instruction hints (*Ihint* for short). The instructions are selected by an analysis algorithm and the cache architecture is simulated.

3.1 The *Ihint* Cache

Figure 1 shows the proposed cache architecture which is a slightly modified version of the traditional one. We mainly extend the ways of data transferred between the processor and the off-chip memory. In comparison to a standard cache design, we allow direct reads by the processor and provide a FIFO for potentially non-blocking writes. In both case the cache is bypassed and its state is untouched. Accesses to the off-chip memory are controlled by the arbiter.

The cache is accessed in case of a hit. In case of a miss, the actions depend on whether a special-instruction has been encountered. A special-instruction always bypasses the data cache and leads to a direct read from off-chip memory or to a write into the FIFO.

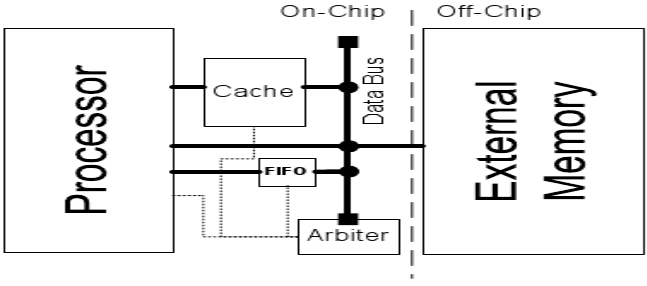


Fig. 1. *Ihint*-Cache Overview

3.2 "Special"-Instruction Selection Process

The hardware provides us with more choices in the memory paths. Now we need to decide which load/store-instructions can be regarded as "special". For this, an analysis algorithm is designed and implemented.

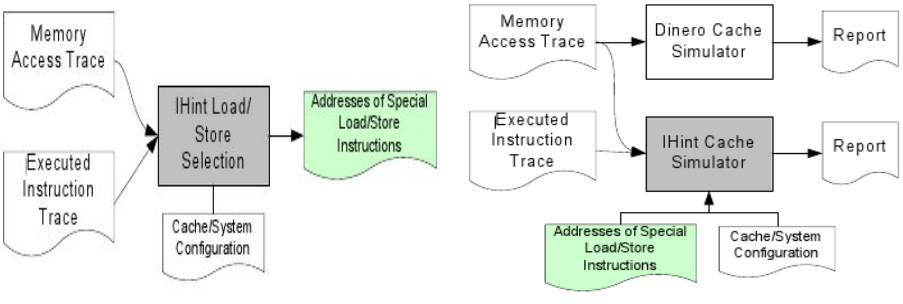


Fig. 2. Instruction selection and cache simulation

The left diagram in Fig. 2 illustrates the workflow of the analysis process. As depicted in the diagram, the analysis algorithm needs a memory access trace and an instruction trace as input. We acquire both trace files with the *ArchC* [2] Instruction Set Simulator. For recording the executed instruction and data access addresses we modified this simulation tool.

The memory access and instruction traces allow us to determine which memory accesses go along with a distinct instruction. This is essential for the analysis algorithm to look for special load/store instructions.

Selecting special instructions is actually to decide whether to move data from off-chip memory to the cache at a read/write miss. The analysis is split into two distinct steps. One for identifying the special loads and one for the stores. The analysis in both steps is based on the difference between the amount of cache hits and misses caused by an individual instruction.

Table 1. Binary instruction break-down by application and architecture

	Instructions						
		MIPS		SPARC		PowerPC	
PEGWIT	total instructions	23149		20652		22405	
	loads/stores	3524	2367	2913	1717	3664	2809
	special loads/stores	41	149	25	80	53	174
CJPEG	total instructions	36043		32008		35388	
	loads/stores	7218	4832	6353	3548	7.171	5.165
	special loads/stores	80	766	67	417	93	821
GSM	total instructions	21899		19888		21124	
	loads/stores	3289	2397	2660	1760	3508	2682
	special loads/stores	18	46	12	38	38	71
DIJKSTRA	total instructions	13251		12100		13753	
	loads/stores	1944	1500	1462	1008	2130	1813
	special loads/stores	11	95	9	101	36	131

For the load-instructions, we keep track which data they have loaded. Basically, this means every word in memory has a pointer to the instruction which has loaded it. Upon a hit we decrease a per instruction miss-counter for the instruction that has previously loaded the requested data at a given address. Upon a read/write miss we increase the counter. Load-instructions with a miss-counter value higher than zero, meaning that the instruction does not introduce any cache hit, are considered as special instructions.

For store-instructions the miss-counter is computed differently. We keep track which instructions have stored data to a particular address in memory. If upon loading a miss is detected, we increase the miss-counter for every previous store-instruction, deploying a decaying penalty for older entries. Currently, we start with a penalty of 10 and halve it for every following older entry until we reach a value of 1. Similar to the load instructions we use a threshold value of zero, meaning that higher values lead to a designation as a special store instruction.

Table 1 shows sample analysis results with four different applications on three different processor architectures. For each application, we counted the instructions in total, the number of load/stores, and the number of special load/store instructions chosen by the analysis algorithm. In general more special store instructions are picked than load instructions and the number of special instructions is not high in contrast to the total number of loads/stores. However, even with this small set of special instructions applications benefit from the *Ihint* approach. In the next section we show the experimental results.

3.3 Cache Simulation

In order to verify the proposed approach as well as to observe the influence of this novel design on performance we developed a cache simulator. Similar to any existing cache models, our tool simulates the basic functionality of the cache

memory and provides statistics on cache hits and misses. The cache configuration can be specified by the user. An additional function of this cache simulator is to process the special load/store instructions. Hence, it requires not only the common memory access trace but also the instruction trace.

The right diagram in Fig. 2 illustrates the simulation process. As shown in the diagram, the simulator takes the memory access trace, the instruction trace, and the addresses of special instructions as input. For each memory access it models the lookup procedure and thereby modifies the status of the cache line. If the memory access is related to a special instruction, the simulator does not load the data into the cache for load operations or puts the data into the FIFO buffer for write operations. During the simulation procedure, the number of cache hits/misses and the amount of data transfers to the main memory are calculated.

For accuracy we adopted the *dinero* [4] cache simulator to verify our own cache simulation component. We compare some report of both simulators to be sure that our component provides correct information.

4 Experimental Results

The *Ihint* concept was verified using four applications of *MediaBench*[11] and *MiBench*[7]. These applications are PEGWIT, CJPEG, GSM, and DIJKSTRA. Each benchmark was run on *MIPS*, *SPARC* and *PowerPC*. PEGWIT deals with the cryptographic key generation; CJPEG handles the *JPEG* encoding; GSM works with audio decoding, and DIJKSTRA computes the shortest path between two nodes in a graph. We measured the amount of off-chip data traffic and the CPU cycles for data accesses. The first metric can be used to evaluate the energy consumption [15], while the second metric represents the performance.

All simulated caches are 4x associative and have 32 byte cache line size. Off-chip memory accesses take 64 cycles for the first access and 32 cycles in burst mode for both reading and writing. Figure 3 depicts the experimental results, where we compare the metric values of *Ihint* with systems without and with caches.

We first observe all diagrams on the left side of the figure for performance issues. Overall, *Ihint* outperforms the normal cache in all benchmarks except for GSM, where both kind of caches behave similarly. This result holds true for all architectures; although they show different values due to the concrete design in instruction sets. Totally, CJPEG achieves an average performance improvement of 78% to the non-cache case and 43% to the normal cache on all three architectures. DIJKSTRAT introduces the best performance, where the *Ihint* cache works 3.3-folds better than the normal cache. With PEGWIT the performance gain is 52% and 72% individually. It is also surprised to see that with this application the cache version requires more time to access the data than the non-cache case. From the corresponding diagram on the right side it can be seen that a large amount of data are transferred from the cache to the off-chip memory. This means that for PEGWIT the additional traffic caused by the cache

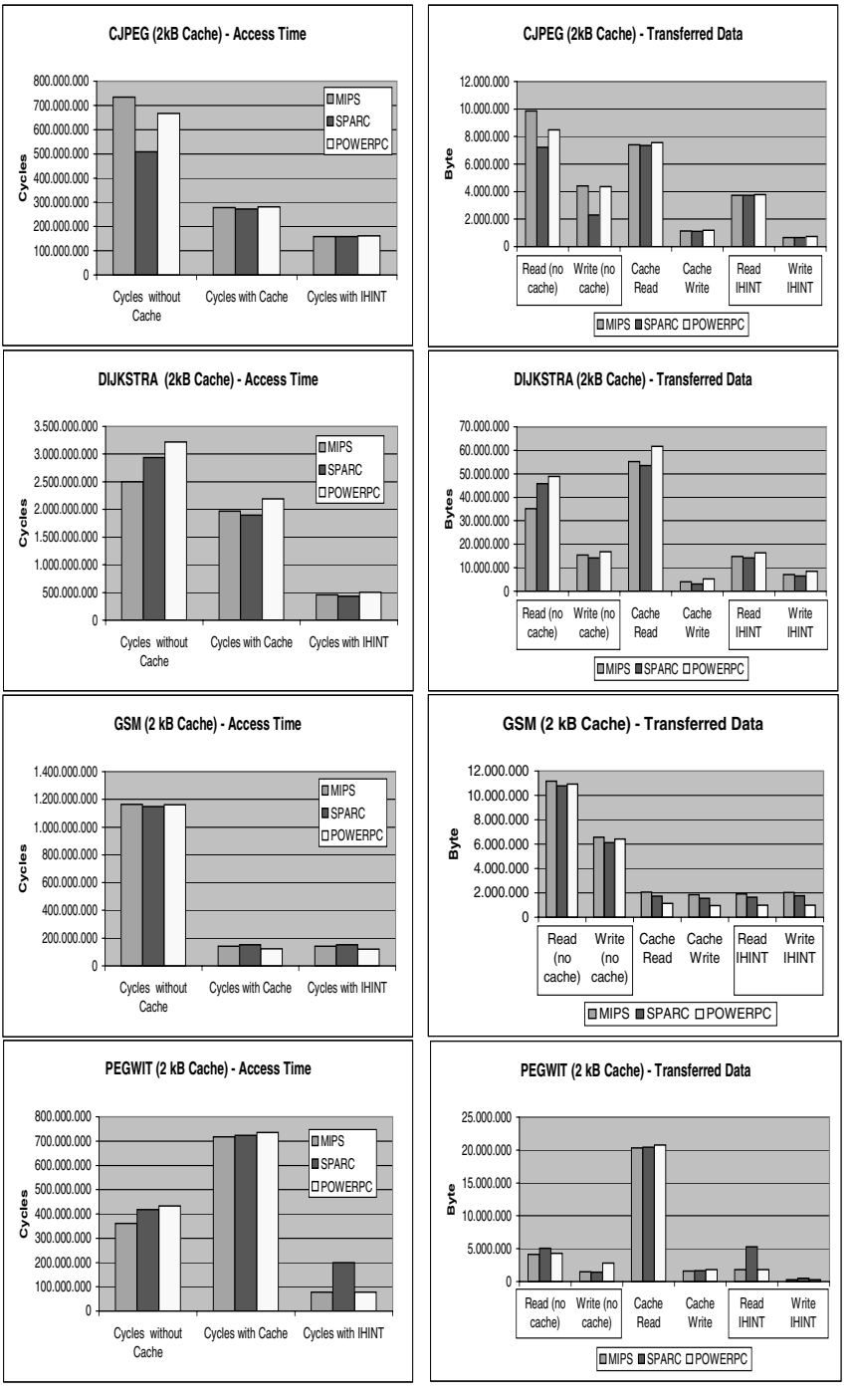


Fig. 3. Experimental results

for implementing the cache policies exceeds the amount of data transfers in the non-cache case. As a result, the introduction of a cache memory does not bring benefit for this application. The *Ihint* cache, however, can better use the cache space, hence showing its advantages. With GSM *Ihint* fails to surpass the cache. This is because the normal cache already performs fairly well with a very low penalty for data access. In this case, the improvement space is limited.

Observing the diagrams on the right side of Fig. 3 similar results with the data traffic can be seen as with the access time: all applications, except GSM, show less data transfers with the *Ihint* approach. The decrease in data traffic indicates a decrease in power requirement. Unfortunately, we do not have an energy model to quantify the amount of power saving.

5 Conclusions

In this paper, we demonstrate an approach of selective bypassing of the cache memory based on instruction hints. It can be seen that our cache extension is able to exploit the new data paths for reading and writing to/from off-chip memory. Our analysis method is successful in selecting suitable load/store instructions at design time. As a result we achieve a much higher efficiency than a normal cache design.

The proposed approach was evaluated on embedded systems with small caches. In the next step, commodity caches with large size and realistic applications will be investigated. It is expected that the same gain can be achieved. In addition, we are also thinking about a hardware approach of selecting the special instructions.

References

1. Abella, J., González, A., Vera, X., O'Boyle, M.: IATAC: a smart predictor to turn-off L2 cache lines. ACM Transactions on Architecture and Code Optimization 2(1), 55–77 (2005)
2. ArchC. Archc - architecture description language (2007)
3. Benitez, D., Moure, J.C., Rexachs, D.I., Luque, E.: Evaluation of The Field-programmable Cache: Performance and Energy Consumption. In: CF 2006: Proceedings of the 3rd conference on Computing frontiers, pp. 361–372 (2006)
4. Edler, J., Hill, M.D.: Dinero IV cache simulator
5. Esakkimuthu, G., Vijaykrishnan, N., Kandemir, M., Irwin, M.J.: Memory System Energy: Influence of Hardware-software Optimizations. In: ISLPED 2000: Proceedings of the 2000 international symposium on Low power electronics and design, pp. 244–246 (2000)
6. Gordon-Ross, A., Vahid, F., Dutt, N.: Fast Configurable-cache Tuning with a Unified Second-level Cache. In: ISLPED 2005: Proceedings of the 2005 international symposium on Low power electronics and design, pp. 323–326 (2005)
7. Guthaus, M.R., Ringenberg, J.S., Ernst, D., Austin, T.M., Mudge, T., Brown, R.B., Mibench: A free, commercially representative embedded benchmark suite. In: 2001 IEEE International Workshop on WWC-4. WWC 2001: Proceedings of the Workload Characterization, pp. 3–14 (2001)

8. Hu, J., Kandemir, M., Vijaykrishnan, N., Irwin, M.J.: Analyzing data reuse for cache reconfiguration. Transactions on Embedded Computing System 4(4), 851–876 (2005)
9. Ishihara, T., Fallah, F.: A Non-uniform Cache Architecture for Low Power System Design. In: ISLPED 2005: Proceedings of the 2005 international symposium on Low power electronics and design, pp. 363–368 (2005)
10. Kandemir, M., Kadayif, I., Choudhary, A., Zambreno, J.A.: Optimizing Inter-nest Data Locality. In: CASES 2002: Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems, pp. 127–135 (2002)
11. Lee, C., Potkonjak, M., Mangione-Smith, W.H.: Mediabench: a tool for evaluating and synthesizing multimedia and communicatons systems. In: MICRO 30: Proceedings of the 30th annual ACM/IEEE international symposium on Microarchitecture, pp. 330–335 (1997)
12. Pingali, V., McKee, S., Hseih, W., Carter, J.: Computation Regrouping: Restructuring Programs for Temporal Data Cache Locality. In: ICS 2002: Proceedings of the 16th international conference on Supercomputing, pp. 252–261 (2002)
13. Sermulins, J., Thies, W., Rabbah, R., Amarasinghe, S.: Cache Aware Optimization of Stream Programs. In: LCTES 2005: Proceedings of the 2005 ACM SIGPLAN/SIGBED conference on Languages, compilers, and tools for embedded systems, pp. 115–126 (2005)
14. Shen, X., Gao, Y., Ding, C., Archambault, R.: Lightweight Reference affinity analysis. In: ICS 2005: Proceedings of the 19th annual international conference on Supercomputing, pp. 131–140 (2005)
15. Sotiriadis, P.P., Chandrakasan, A.P.: A bus energy model for deep submicron technology. IEEE Trans. Very Large Scale Integr. Syst. 10(3), 341–350 (2002)
16. Yang, C., Lee, C.: HotSpot Cache: Joint Temporal and Spatial Locality Exploitation for I-cache Energy Reduction. In: ISLPED 2004: Proceedings of the 2004 international symposium on Low power electronics and design, pp. 114–119 (2004)

A Holistic Approach for Performance Measurement and Analysis for Petascale Applications

Heike Jagode[1,2], Jack Dongarra[1,2], Sadaf Alam[2], Jeffrey Vetter[2], Wyatt Spear[3], and Allen D. Malony[3]

[1] The University of Tennessee
[2] Oak Ridge National Laboratory
[3] University of Oregon
{jagode,dongarra}@eecs.utk.edu
{alamsr,vetter}@ornl.gov
{wspear,malony}@cs.uoregon.edu

Abstract. Contemporary high-end Terascale and Petascale systems are composed of hundreds of thousands of commodity multi-core processors interconnected with high-speed custom networks. Performance characteristics of applications executing on these systems are a function of system hardware and software as well as workload parameters. Therefore, it has become increasingly challenging to measure, analyze and project performance using a single tool on these systems. In order to address these issues, we propose a methodology for performance measurement and analysis that is aware of applications and the underlying system hierarchies. On the application level, we measure cost distribution and runtime dependent values for different components of the underlying programming model. On the system front, we measure and analyze information gathered for unique system features, particularly shared components in the multi-core processors. We demonstrate our approach using a Petascale combustion application called S3D on two high-end Teraflops systems, Cray XT4 and IBM Blue Gene/P, using a combination of hardware performance monitoring, profiling and tracing tools.

Keywords: Performance Analysis, Performance Tools, Profiling, Tracing, Trace files, Petascale Applications, Petascale Systems.

1 Introduction

Estimating achievable performance and scaling efficiencies in modern Terascale and Petascale systems is a complex task. A diverse set of tools and techniques are used to identify and resolve their performance and scaling problems. In most cases, it is a challenging combination of tasks which include porting the various software components to the target systems, source code instrumentation usually associated with performance tests and management of huge amounts of performance measurement data. The complexity of analysis concepts is extensively described in [1]. In order to address these issues, we propose a methodology for performance measurement and analysis that is aware of applications and the underlying system hierarchies. On the application level, we measure cost distribution and runtime dependent values for different components of the underlying programming model. On the system front, we measure and analyze

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 686–695, 2009.

information gathered for unique system features, particularly shared components in the multi-core processors.

A detailed analysis of the massivly parallel DNS solver - S3D - on the latest generation of large scale parallel systems allows a better understanding of the complex performance characteristics of these machines. As part of the Petascale measurement effort, we target a problem configuration capable of scaling to a Petaflops-scale system. A complete performance analysis approach requires studying the parallel application in multiple levels such as the computation level, communication level, as well as cache level. This multilayered approach makes it a challenging exercise to monitor, analyze and project performance using a single tool. Hence, a combination of existing performance tools and techniques have been applied to identify and resolve performance and scaling issues. In this context, we are evaluating the collected performance data of S3D using the PAPI library [2] as well as TAU [3], VampirTrace [4], and Vampir [5] toolsets for scalable performance analysis of Petascale parallel applications. Such tools provide detailed analyses that offer important insight into performance and scalability problems in the form of summarized measurements of program execution behavior.

The report is organized as follows: first we provide a brief description of the target systems' features. This is followed by a summary of the applied performance analysis tools. A brief outline of the high-end scientific application that is targeted for this study is provided at the end of section 2. In section 3, we provide extensive performance measurement and analysis results that are collected on the Cray XT4 as well as IBM BlueGene/P system using a set of scalable performance tools. We then provide an insight into the factors resulting in the load imbalance among MPI tasks that cause signifcant idle time for the S3D application runs on a large number of cores. Finally, we outline conclusions and a list of future plans.

2 Background

2.1 Computer Systems

We start with a short description of the key features, most relevant for this study, of the super computer systems that had the following characteristics in December 2008. The Jaguar system at Oak Ridge National Laboratory (ORNL) is based on Cray XT4 hardware. It utilizes 7,832 quad-core AMD Opteron processors with a clock frequency of 2.1 GHz and 8 GBytes of memory (maintaining the per core memory at 2 GBytes). Jaguar offers a theoretical peak performance of 260.2 Tflops/s and a sustained performance of 205 Tflops/s on Linpack [6]. In the current configuration, there is a combination of processing nodes with DDR-667 and DDR-800. Peak bandwidth of DDR-800 is 12.8 GBytes/s. If needed, especially for benchmarking purposes, a user has an opportunity to specify memory bandwidth requirements at the job submission. In this paper we merely use nodes with 800 MHz memory. The nodes are arranged in a 3-dimensional torus topology of dimension $21 \times 16 \times 24$ with full SeaStar2 router through HyperTransport. The network offers toroidal connections in all three dimensions.

The Argonne National Laboratory (ANL) Intrepid system is based on IBM BlueGene/P technology. The system offers 163,840 IMB PowerPC 450 quad-core processors with a clock frequency of 850 MHz and 2 GBytes of memory (maintaining the per core memory at 500 MBytes). The peak performance of the system is 557 Tflops/s and

the sustained Linpack performance is 450.3 Tflops/s [6]. The processors are arranged on a torus network of maximum dimension $40 \times 32 \times 32$, which can be divided into sub-tori - as one example, the smallest torus is of dimension $8 \times 8 \times 8$ - or even smaller partitions with open meshes. An individual partition can not be shared between several users. In contrast to the Cray XT architecture, the BlueGene/P offers the user full control over how the tasks are placed onto the mesh network. At the time of the experiments, hardware counter metrics, such as cache misses and floating point operations were not immediately available on BG/P because of an incomplete PAPI implementation on the platform.

2.2 Performance Analysis Tools

Before we show detailed performance analysis results, we will briefly introduce the main features of the used profiling and tracing tools that are relevant for this paper.

The TAU Performance System is a portable profiling and tracing toolkit for performance analysis of parallel programs [3]. Although it comes with a wide selection of features, for this paper it is mainly used to collect performance profile information through function and loop level instrumentation. TAU profiling collects several metrics for each instrumented function or section of code. Here we focus on the exclusive values. These provide the time or counter value accrued in the body of a function over the program's execution and exclude the time or value accrued in the function's soubroutines. TAU also provides a number of utilities for analysis and visualization of performance profiles.

To analyze details of the S3D application, event-based program traces have been recorded using VampirTrace [4]. The generated *Open Trace Format* (OTF) [7] trace files have then been analyzed using the visualization tool Vampir [5]. Vampir is a tool for performance and optimization that enables application developers to visualize program behavior at any level of detail. For this paper, a considerable amount of performance measurement data has been produced. For that reason, the huge data volume has been analyzed with the parallel version of Vampir [5].

2.3 Application Case Study: Turbulent Combustion (S3D)

The application test case is drawn from the workload configurations that are expected to scale to large number of cores and that are representative of Petascale problem configurations. S3D is a massively parallel DNS solver developed at Sandia National Laboratories. Direct numerical simulation (DNS) of turbulent combustion provides fundamental insight into the coupling between fluid dynamics, chemistry, and molecular transport in reacting flows. S3D solves the full compressible Navier-Stokes, total energy, species, and mass continuity equations coupled with detailed chemistry. The governing equations are solved on a conventional three-dimensional structured Cartesian mesh. The code is parallelized using a three-dimensional domain decomposition and MPI communication. Ghost zones are constructed at the task boundaries by non-blocking MPI communication among nearest neighbors in the three-dimensional decomposition. Time advance is achieved through a six-stage, fourth-order explicit Runge-Kutta method [8]. S3D is typically run in weak-scaling mode where the sub-grid size remains constant per computational thread.

3 Measurements and Analysis Methodology

As indicated earlier, one of the key goals of this study is to understand and categorize the application behavior on Teraflops-scale leadership computing platforms. In order to achieve this goal, we propose a methodology for performance measurement and analysis that is aware of applications and the underlying system hierarchies as well as the underlying programming model adopted by the application and implemented by the system. We systemetically gathered and analyzed data that enabled us to understand the control flow of critical computation and communication phases, utilization of system features in terms of hardware counter data and behavior of MPI communication operations. The next subsections summarize results collected from a set of profiling, tracing and analysis tools.

3.1 Hardware Event Data Profile

To generally measure the computational efficiency of the S3D application, the ratio of instructions executed per CPU cycle (IPC) has been computed using the following PAPI native events [2]: `IPC = RETIRED INSTRUCTIONS / CPU CLOCKS NOT HALTED`

All the performance analysis utilities summarized in section 2.2 support PAPI counter data. The TAU performance system was used to collect these events. For the hardware performance event measurements, S3D was run on 64 processors on Jaguar. IPC indicates the degree to which the hardware is able to exploit instruction level parallelism in the S3D program [9]. Figure 1 presents the effective instructions per CPU cycle, computed for the 13 most time consuming functions in S3D. Higher is better and low IPC may indicate the presence of performance issues. The AMD Opteron processor can issue up to three AMD64 instructions per cycle and per core [10]. A comparison of the measured IPC with the maximum number reveals the low processor efficiency for most of the time consuming functions.

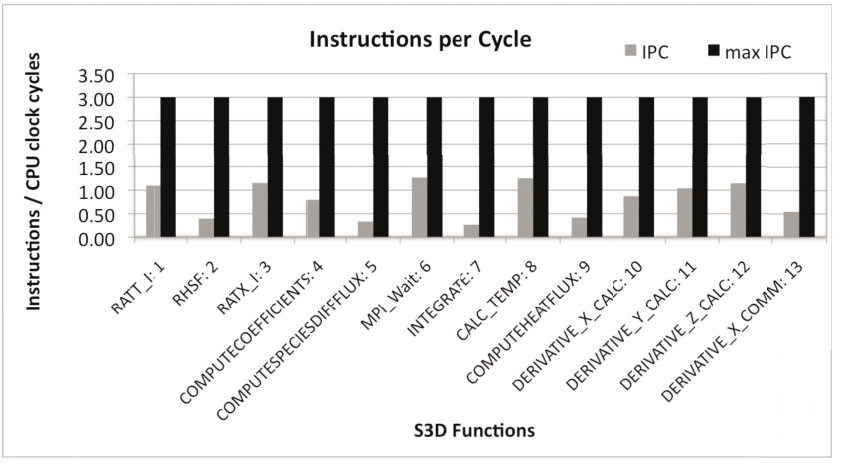


Fig. 1. Instructions per cycle (IPC) for the 13 most time consuming S3D functions (mean)

Table 1. S3D total execution time (s)

Architecture	VNM	SMP
Jaguar	813 s	613.4 s
BG/P	2920.41 s	2918.99 s

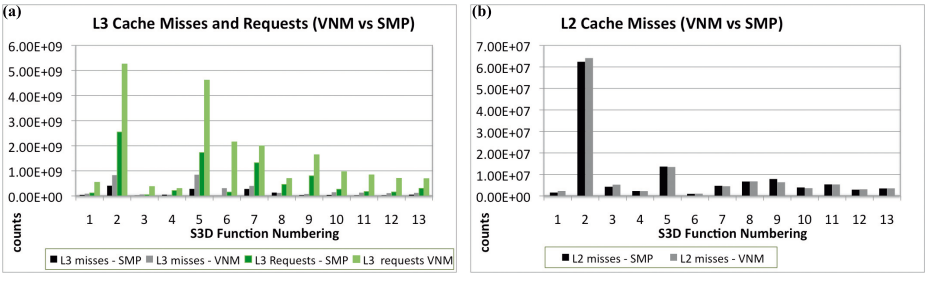


Fig. 2. (a) L3 cache misses and requests (mean); (b) L2 cache misses (mean)

In order to investigate causes of low IPC, we quantitatively evaluate the performance of the memory sub-system. For that purpose the application was run in SMP (1 core per node) as well as VN mode (4 cores per node). We see a significant slowdown of 25% in VN mode as compared to single-core mode on Jaguar only. Table 1 shows the total execution time for runs on Jaguar and BG/P using the two different modes. We do not see such a serious performance discrepancy on BG/P. Note, that BG/P also utilizes quad-core nodes but with PowerPC 450 processors that offer a much lower clock frequency (850 MHz) which results in a longer execution time. On both systems - Cray XT4 and BG/P - the L3 cache is shared between all four cores. We collected hardware performance events using the PAPI library [2] that confirms our findings. L3 cache requests are measured and computed using the following PAPI native events:

```
L3 REQUESTS = READ REQUESTS TO L3 + L3 FILLS CAUSED BY L2 EVICTION
```

Note: In VNM all L3 cache measurements have been divided by 4 (4 cores per node on Jaguar)

Figure 2 (a) depicts the number of L3 cache misses and requests when using 4 cores versus 1 core per node for the 13 most expensive functions of the S3D application. It appears that the performance degradation in VN mode is due to the L3 cache behavior. In VN mode we see roughly twice as many L3 cache requests and misses compared to SMP mode. It is not surprising that L3 cache misses increase with VN mode since if every thread is operating on different data, then one thread could easily evict the data for another thread if the sum of the 4 working threads is greater than the size of the L3 cache. However, the increase of L3 requests is rather questionable. The L3 cache serves as a victim cache for L2. In other words, if data is not in L2 cache then L2 TLB checks the L3 cache which results in a L3 request. As mentioned earlier the L3 cache is shared between all four cores while the L2 cache is private. Based on this workflow, it is not clear why the number of L3 requests increases so dramatically when using all 4 cores per node. As verification we measure the L2 cache misses in SMP and VN mode and Fig. 2 (b) presents the comparison. It clearly shows that the number of L2 cache misses

does not increase when all four cores are used compared to SMP mode. All the more it is a moot point where the double L3 cache requests come from when VN mode is used. Note that Fig. 2 (a) and (b) use S3D function numbering only while in Fig. 1 the numbers are associated with the corresponding name of the S3D functions.

Recent discussions with the research lead [1] for the PAPI project have led us to wonder if this is an artifact of the measurement process. The L3 events in AMD Opteron quad-core processors are not monitored in four independent sets of hardware performance registers but in a single set of registers not associated with a specific core (often referred to as "shadow" registers). There are independent counter registers on each core for most performance events. When an L3 event is programmed into one of these counters on one of these cores, it gets copied by hardware to the shadow register. Thus, only the last event to be programmed into any core is the one actually measured by all cores. When several cores try to share a shadow register, the results are not clearly defined. Performance counter measurement at the process or thread level relies on the assumption that counter resources can be isolated to a single thread of execution. That assumption is generally no longer true for resources shared between cores - like the L3 cache in AMD quad-core nodes. New methods need to be developed to appropriately collect and interpret hardware performance counter information collected from such multi-core systems with interesting shared resources. This is an open area of on-going research.

3.2 Tracing Analysis Using Vampir

The results that have been presented so far show aggregate behavior or profile during the appliction run and does not provide an information about the time line of these operations. Unlike profiling, the tracing approach records function calls, messages, etc. as timed events; that is as a combination of timestamp, event type, and even specific data [1]. Tracing experiments allow users detailed observations of their parallel application to understand a time dependent behavior of the application run. However, the tracing approach also indicates the production of a very large protocol data volume. This is a good time to mention once more the advantage of using a mixture of profiling and tracing tools to overcome those obstacles. Using a profiler prior to the application of a trace tool, for example, can facilitate the exclusion of functions which have a high call frequency [1] and low inclusive time per call. Otherwise, those trivial routines take up most of the time for accessing system clock and maintaining callstack information which makes its timing less accurate. On this account, the TAU profiling tool has been used to create a list of S3D functions that can be excluded from tracing. Turning off those frequently called low level functions results in a S3D trace of a reasonable size while the data accuracy is maintained.

For recording and analyzing event-based program traces, the Vampir suite - a widely recognized tracing facility - has been used. The suite itself is briefly summarized in section 2.2. For the performance analysis via VampirTrace and Vampir, the S3D application has been run on 512 processors in VN mode on Jaguar. Figure 3 (a) presents the number of processes that are actively involved in a given activity at a certain point in time. This information is shown as a vertical histogram. It can be identified how the

[1] Dan Terpstra is the research lead for the PAPI project: terpstra@eecs.utk.edu

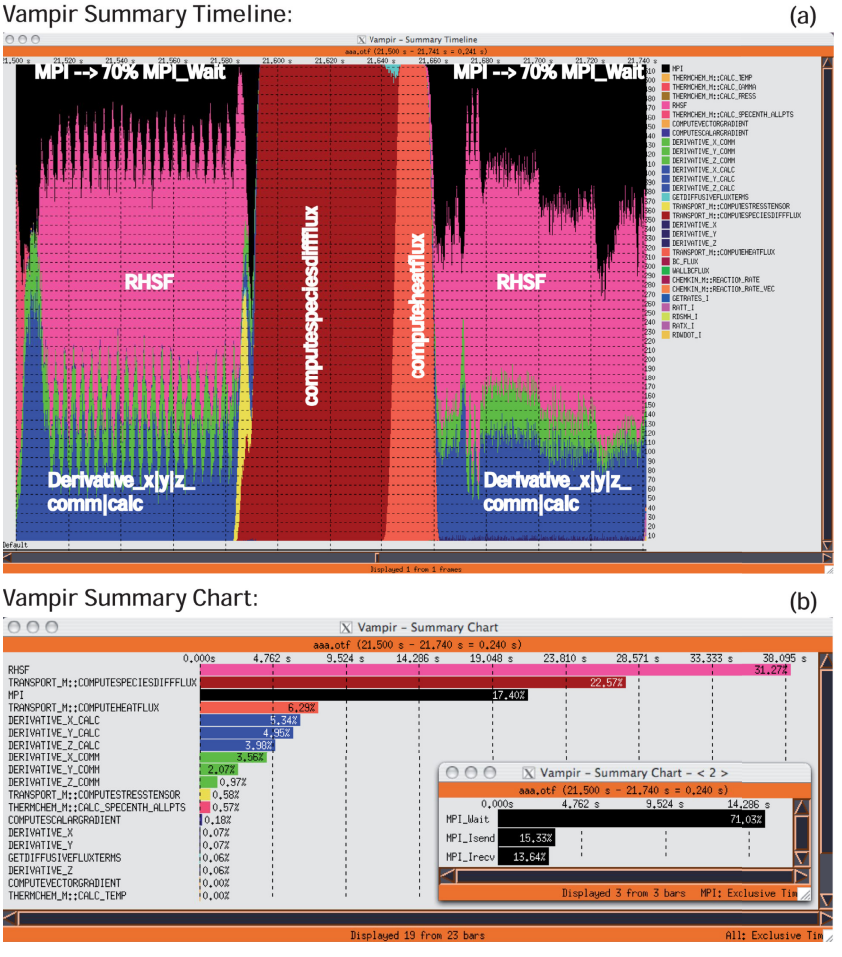


Fig. 3. S3D performance analysis via Vampir

computation and communication is distributed for a sequence of operations that are repeated in x-, y-, and z-dimensions in one logical time step of the S3D application. Even with visualization tools that graphically present performance data - like TAU and Vampir - for the manual analysis approach some expertise in the field of parallel computing is required to identify possible performance problems [1]. From the high level perspective shown in Fig. 3 (a) together with Fig. 3 (b), it appears that computational work in S3D is not well-balanced in its distribution between the computing resourses. The Summary Chart (Fig. 3 (b)) gives an overview of the accumulated time consumption across all processes and activities. We split MPI times up so that we can see that more than 70% of the entire MPI time is spent in `MPI_Wait`. The rest of the MPI time is spent in non-blocking send and receive operations. Our present main intent is to identify the root causes of load imbalance on large-scale runs and reduction in performance efficiencies on multi-core processors.

3.3 Weak-Scaling Results and Topology Effects on BlueGene/P

Using TAU we collected data from S3D on BlueGene/P for jobs ranging from 1 to 30,000 cores. From this weak-scaling study it was apparent that time spent in communication routines began to dominate as the number of cores increased. A runtime breakdown over trials with increasing numbers of cores, shown in Fig. 4, illustrates this phenomenon. In the 30,000 core case the time spent in routines `MPI_Barrier`, `MPI_Wait` and `MPI_Isend` rose as high as 20.9, 12.7 and 8.7 percent respectively while the time spent in compute routines was not significantly different from lower processor counts.

We further observed deviation between individual threads in time spent in communication routines. The pattern of deviation suggested a load imbalance impacted by node topology. We tested this hypothesis by running an 8000 core test with a random node mapping replacing the default. The default trial had a runtime of 60 minutes and 26.4 seconds. The random node map decreased the runtime to 56 minutes and 47.4 seconds, a speedup of approximately 6%. The change in runtime was almost entirely the result of MPI behavior. The random map saw an average per-thread increase in the `MPI_Wait` routine from 202.3 to 297.4 seconds. However time in `MPI_Barrier` dropped from 349.2 to 48.5 seconds.

The results from the random mapping test indicate that it is possible to improve significantly over BlueGene/P's default mapping for the S3D application. We are presently investigating alternative mapping schemes to find an optimal topology for S3D on the platform.

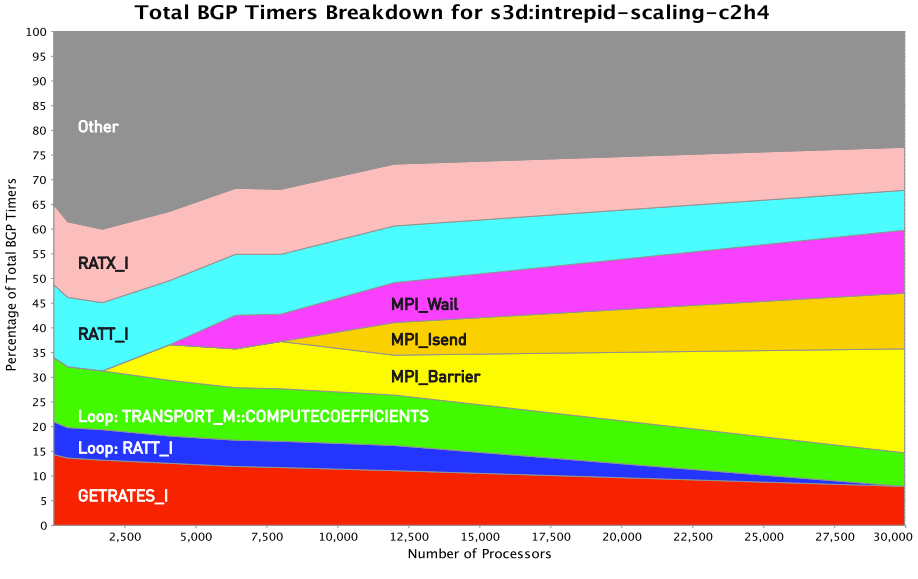


Fig. 4. S3D Scaling Study on BlueGene/P

4 Conclusion and Future Plans

This paper proposes a methodology for performance measurement and analysis that is aware of applications as well as high-end Terascale and Petascale system hierarchies. This approach is demonstrated using a Petascale combustion application called S3D on two high-end systems, Cray XT4 and IBM Blue Gene/P. Our way shows that using a mixture of profiling and tracing tools is highly advisable. It provides important insight into performance and scalability problems by aggregating behavior indicated in profiles with temporal information from the execution time line. This performance analysis approach allowed us to identify a major load imbalance issue when the number of cores are increased. More than 70% of the communication time is spent in `MPI_Wait` on large-scale runs. Our present main intent is to identify the root causes of this behavior.

A deeper investigation of the derivation of time spent in communication routines for individual processes on the Blue Gene architecture shows that the load imbalance is impacted by the node mapping pattern. Using a random instead of the default node mapping confirms the finding and yields a significant performance improvement of about 6% for the entire S3D application. Beside random mapping patterns, we are currently investigating alternative mapping schemes to find an optimal topology for S3D on the Blue Gene/P platform.

Our data collection of hardware performance events shows questionable findings for L3 cache behavior on AMD Opteron processors if all 4 cores on a node are actively used. While only one core can monitor L3 events, it is not clear what happens when several cores try to share the single set of hardware performance registers that is provided to monitor L3 events. Conflicts in measuring those events related to resources that are shared between cores indicate the need for further research on a portable hardware counter interface for multi-core systems. It is a focus of on-going research in the Innovative Computing Laboratory of the University of Tennessee to develop methods to address this issue.

Acknowledgements

The authors would like to thank the PAPI and Vampir team for their great support. Furthermore, Philip Roth (ORNL) is greatly acknowledged for providing a working S3D version for the BG/P architecture. This research was sponsored by the Office of Mathematical, Information, and Computational Sciences of the Office of Science (OoS), U.S. Department of Energy (DoE), under Contract No. DE-AC05-00OR22725 with UT-Battelle, LLC as well as by the University of Oregon DoE grant from the OoS under Contract No. DE-FG02-07ER25826. This work used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the Department of Energy under Contract DE-AC05-00OR22725 and of the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract AC02-06CH11357. These resources were made available via the Performance Evaluation and Analysis Consortium End Station, a Department of Energy INCITE project.

References

1. Brunst, H.: Integrative Concepts for Scalable Distributed Performance Analysis and Visualization of Parallel Programs, Ph.D Dissertation, Shaker Verlag (2008)
2. PAPI Documentation: `http://icl.cs.utk.edu/papi`
3. TAU User Guide: `www.cs.uoregon.edu/research/tau/docs/newguide/index.html`
4. Jurenz, M.: VampirTrace Software and Documentation, ZIH, TU Dresden: `http://www.tu-dresden.de/zih/vampirtrace`
5. VampirServer User Guide: `http://www.vampir.eu`
6. Top500 list: `http://www.top500.org`
7. Knüpfer, A., Brendel, R., Brunst, H., Mix, H., Nagel, W.E.: Introducing the Open Trace Format (OTF). In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3992, pp. 526–533. Springer, Heidelberg (2006)
8. Kennedy, C.A., Carpenter, M.H., Lewis, R.M.: Low-storage explicit Runge-Kutta schemes for the compressible Navier-Stokes equations. Applied numerical mathematics 35(3), 177–264 (2000)
9. Drongowski, P.: Basic Performance measurements for AMD Athlon 64 and AMD Opteron Processors (2006)
10. Software Optimization Guide for AMD Family 10h Processors, Pub. no. 40546 (2008)

A Generic and Configurable Source-Code Instrumentation Component

Markus Geimer[1], Sameer S. Shende[2], Allen D. Malony[2], and Felix Wolf[1,3]

[1] Jülich Supercomputing Centre
Forschungszentrum Jülich, Germany
{m.geimer,f.wolf}@fz-juelich.de
[2] Department of Computer and Information Science
University of Oregon, Eugene, OR, USA
{sameer,malony}@cs.uoregon.edu
[3] Department of Computer Science
RWTH Aachen University, Germany

Abstract. A common prerequisite for a number of debugging and performance-analysis techniques is the injection of auxiliary program code into the application under investigation, a process called *instrumentation*. To accomplish this task, source-code preprocessors are often used. Unfortunately, existing preprocessing tools either focus only on a very specific aspect or use hard-coded commands for instrumentation. In this paper, we examine which basic constructs are required to specify a user-defined routine entry/exit instrumentation. This analysis serves as a basis for a generic instrumentation component working on the source-code level where the instructions to be inserted can be flexibly configured. We evaluate the identified constructs with our prototypical implementation and show that these are sufficient to fulfill the needs of a number of todays' performance-analysis tools.

1 Introduction

As a prerequisite for various performance-analysis and debugging techniques, it is often necessary to insert additional code fragments into the application that is currently under investigation, e.g., to validate parameters given to a function call, read hardware counter values such as the number of cache misses, or query the system clock to calculate the time spent in a certain code region. This process of adding extra code to be executed at runtime is called *instrumentation* and can be accomplished in a number of different ways.

A well-accepted technique of instrumenting an application is the so-called *source-code instrumentation* method, which is the subject matter of this paper. With this approach, additional code fragments such as function calls are directly inserted into the application's source code at appropriate places before compilation. Although this can be done manually by the developer—being quite time-consuming and error-prone—it is generally more convenient to perform this step automatically using a source-code preprocessor. Since instrumentation is entirely performed on the source-code level, its granularity can be flexibly controlled and is even not restricted to functions, but can also be done, e.g., for program phases, basic blocks, loops or individual statements. In

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 696–705, 2009.

addition, correlating analysis results gained from such an instrumentation with locations in the source code is trivial. And finally, this approach is platform-independent as a source-code preprocessor can be implemented in a very portable way.

Unfortunately, to the authors' knowledge, none of the source-code instrumentation tools available today is flexible enough to satisfy the need of the tool developer community for a generic instrumentation component, since the commands to be inserted into the application's source are typically hard-coded for a particular purpose or toolset. To overcome this situation, this paper investigates the general requirements for such a configurable source-code instrumentor. As a starting point, our initial focus is primarily on instrumenting routine entries and exits, a feature which is commonly needed by performance-analysis and debugging tools. Not to start entirely from scratch, our prototypical implementation used to evaluate the identified constructs is based on the instrumentor of the TAU performance-analysis framework [1].

The remainder of this paper is structured as follows: after a review of related work in Section 2, we summarize the architecture of the aforementioned TAU instrumentor in Section 3. Section 4 then discusses the requirements of a configurable instrumentation component and the basic constructs that we identified as "building blocks" for user-defined instrumentation. Next, Section 5 evaluates the presented configuration concepts by mapping the manual instrumentation API of various performance-analysis toolsets onto the generic constructs, before we conclude the paper and outline directions of future work in Section 6.

2 Related Work

A simple way of inserting instrumentation code into an application is specified by the Message Passing Interface (MPI) standard [2]. Here, all library calls also exist with a second entry point name using the `PMPI_` prefix, allowing a user or tool developer to provide an interposed wrapper library intercepting `MPI_` calls issued by the user code. However, this approach can only capture the behavior of the instrumented MPI routines and has to be used in conjunction with one or more of the techniques described below to also gain insights into the computational core of the application.

Somewhat similar interfaces for instrumenting communication-related events are provided by the PERUSE MPI extension [3] as well as the GASP performance-analysis tool interface [4] targeting partitioned global address space (PGAS) languages. In both cases, the user of these interfaces is given the possibility to register callback functions for events of interest. Although providing very detailed information about the internals of the communication, pure user functions are still ignored.

A complementary approach applicable to user code is to leverage the capability provided by many of todays' compilers to automatically instrument the entry and exit points of functions. Although this sounds like a convenient way to instrument user code, this approach has several drawbacks. First, this feature sometimes has to rely on undocumented or unsupported compiler functionality (e.g., for the IBM xl compilers). Second, it is absolutely compiler-dependent whether instrumentation is performed before or after code optimization, i.e., the granularity of the results may differ significantly when switching between compilers. And third, the user has only very limited control over

what is instrumented. Enabling or disabling the instrumentation on a per-file level is of course always possible, but control on the function level is only supported by few compilers, typically using relatively inconvenient command-line interfaces [5].

In contrast to compiler-based instrumentation, the binary instrumentation technique [6,7] inserts measurement calls after the program's binary code is generated. In this case, the additional instrumentation code is injected either at runtime by patching the application's binary code in memory, or through rewriting the application executable prior to execution. However, this low-level technique is very architecture- and compiler-dependent, which restricts its applicability to the supported set of platforms/compilers. In addition, it suffers from a non-negligible runtime overhead, since calls to the inserted instrumentation code typically cannot be performed directly but have to go through some sort of indirection (e.g., using so-called trampolines). Nonetheless, this technique is the only choice if the application's source code is not available.

As an example of a source-code preprocessing tool, OPARI [8] specifically focuses on instrumenting OpenMP directives, requiring it to be used in conjunction with some other technique to instrument user functions. This could be done, e.g., using the aforementioned TAU source-code instrumentor, which forms the basis of our prototypical implementation and will therefore be covered in more detail in the next section.

An alternative framework that can be used to write source-to-source translation tools is ROSE [9]. Although being very powerful through the ability of regenerating source code after modifying the abstract syntax tree im memory, ROSE is currently only distributed for x86 and x86-64 architectures, limiting the portability of tools written on top of it.

3 TAU Source-Code Instrumentor Overview

Altering the source code of an application by a preprocessor before it is passed to the compiler typically involves parsing the source code to infer the locations of potential instrumentation points. To relieve developers of source-to-source translators from the burden of writing their own parsers and to support the development of such tools, the TAU project has developed the Program Database Toolkit (PDT) [10]. As depicted in Figure 1, PDT consists of several components that are used in different steps of the instrumentation workflow described below.

The first step is to parse the source-code files using commercial-grade compiler front-ends which build an internal representation in form of an abstract syntax tree and write this information to an intermediate language (IL) file. Next, IL analyzers walk the abstract syntax tree stored in the IL file and extract a reasonable subset of the syntactic entities, storing the result in a program database (PDB) ASCII text file. PDB files provide information such as the list of all input files read, a list of all routines including the source-code locations of their declaration and definition, and a list of all statements for each routine, again providing their source-code locations. To simplify tool development, PDT also provides a C++ class library (DUCTAPE) as a convenient interface to access the PDB data.

The TAU source-code instrumentor, built on top of the DUCTAPE library, first reads the generated PDB file, analyzes the contained syntactic information and generates a list

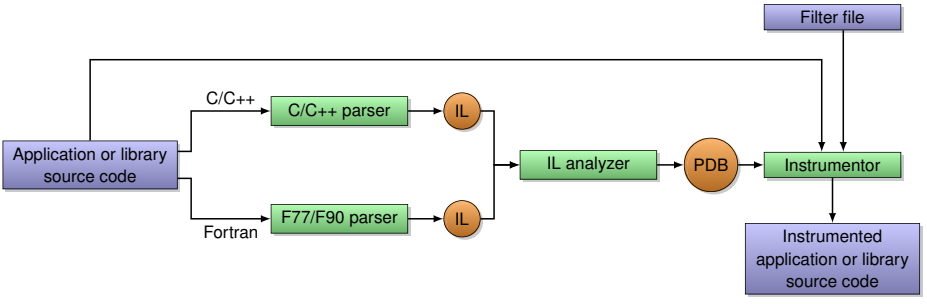


Fig. 1. Overview of the TAU instrumentor workflow. The source code is first processed by a parser front-end for the corresponding programming language, generating an intermediate language (IL) file. This file is then converted by an IL analyzer into an program database (PDB) file. The instrumentor itself then reads the PDB file, the application or library source code as well as a filter file and generates the modified, instrumented source.

of *instrumentation requests*. At this point, filter rules specified in a configuration file given to the instrumentor are applied to selectively enable or disable instrumentation for certain code regions (e.g., functions or loops). Finally, the original application or library source code is read line by line and augmented with calls to the TAU measurement library according to the instrumentation requests that remain after filtering.

The TAU source-code instrumentor currently works with C, C++, and Fortran. It is robust, can process very large source files, and is able to instrument routines, methods, and loops. However, the instrumentor only generates TAU measurement code. A general approach would allow any measurement library to be used and would be applicable to other languages such as Java or the emerging HPCS languages X10, Chapel and Fortress, provided that suitable parser front-ends are available.

4 Requirements for a Configurable Instrumentation Component

To generalize the instrumentor to be used with performance-analysis or debugging tools other than TAU, the hard-coded insertion of calls to the TAU measurement API has to be replaced by the injection of arbitrary code fragments that can be specified by the user (i.e., typically the developer of the corresponding tool). In this context, the most important questions to be considered are:

- What are the basic constructs needed to specify a user-defined instrumentation?
- Which additional information available at instrumentation time might be useful?
- How can this information be referenced in the users' code fragments?

The following subsections examine these questions in more detail and present our current solution. They are structured based on the basic constructs that we have identified as the "building blocks" for user-defined instrumentation. All examples are given in the syntax used by our current prototype implementation based on the TAU instrumentor.

4.1 Entering a Routine

One important point where tool developers typically want to insert extra instrumentation code is at the begin of functions, to be notified when the routine is entered. For this type of instrumentation request, the `entry` construct is provided:

```
entry file="*" routine="#" code="printf(\"Entering\\n\");"
```

To make this construct generic and flexible, both file and routine names can be specified using wildcards. Note that we are using the hash character (#) as a wildcard for routine names, since the asterisk (*) can be part of the function signature in C/C++ and we wanted to avoid introducing an escape character for routine names. In addition, since such "catch all" rules as shown in the example are commonly used, we decided to make them the default behavior, allowing the user to omit the `file` and `routine` parts of the specification line. For the specified code fragment, we adopted the standard C syntax to quote special characters within strings using the backslash character, also supporting line breaks (\n) and tabulators (\t). This allows for inserting multiple code lines with a single specification rule. Alternatively, several `entry` clauses for the same file/line combination can be given since their code fragments will be concatenated in the order of appearance in the specification file, separated by a line break. Of course, all filter rules defined in a filter file given to the instrumentor still apply, allowing generic specification rules which are then not used for certain files or routines.

To leverage "instrumentor knowledge" in the code snippets to be inserted, a number of textual substitutions are performed just before they are written to the output file. For example, the keyword `@FILE@` will be replaced by the name of the input file and `@LINE@` by the line number in the original source file at which the code is inserted. Although this information is in principle also available through the C preprocessor macros `__FILE__` and `__LINE__`, using these macros will usually insert the wrong values since the file that is actually being compiled will be the instrumented source file with a temporary name as well as displaced code lines due to the instrumentation. Theoretically, this can be corrected using `#line` directives, however, adding them correctly is non-trivial. Moreover, the `__FILE__` and `__LINE__` macros cannot be used inside of strings.

Besides file and line number information, the instrumentor can also provide the name of the routine (i.e., the full signature) as well as the line and column of both begin and end of the function body. Table 1 provides a full list of all keyword substitutions that we deemed useful and that are currently supported.

4.2 Leaving a Routine

Similar to the point of entering a routine, the location where the routine is left is another important point to insert instrumentation code. This applies to the end of the function body as well as to every intermittent `return` statement. For this purpose, the `exit` construct is provided:

```
exit file="*" routine="#" code="printf(\"Leaving\\n\");"
```

Again, the same wildcard, quoting and keyword substitution rules as described in the context of the `entry` construct (Sec. 4.1) apply. Note that it is possible to distinguish

Table 1. Keyword substitutions performed while inserting user-defined code fragments

Keyword	Substitution
All constructs:	
`@FILE@`	File name
`@LINE@`	Source line of insertion
`@COL@`	Column of insertion
`decl`, `init`, `entry`, `exit`, `abort` only:	
`@ROUTINE@`	Routine name
`@BEGIN_LINE@`	Begin line of routine body
`@BEGIN_COL@`	Begin column of routine body
`@END_LINE@`	End line of routine body
`@END_COL@`	End column of routine body
`decl`, `entry`, `exit`, `abort` only (C++):	
`@RTTI@`	Dynamic routine name (class/member function templates)
`init` only (C/C++):	
`@ARGC@`	Name of first paramater to `main()`
`@ARGV@`	Name of second parameter to `main()`

different `return` statements of a routine using the `@LINE@` keyword substitution, which might be handy for debugging purposes.

For C and C++, the expression after the `return` keyword defining the return value can be arbitrarily complex. To insert the `exit` code fragment as late as possible (e.g., for accurate time measurements), the source-code needs to be slightly rewritten. First, the result of the return expression is assigned to a local variable. Next, the given `exit` code snippet is inserted and finally, the `return` statement returning the value of the aforementioned local variable is generated. Note that replacing a single-line expression with multiple lines of code might require the creation of a new $\{\ldots\}$ block in C/C++ or modifying the surrounding `if` statement in Fortran.

4.3 Variable Declarations

The code fragments specified by a user to instrument routine entries and exits might require the declaration of local variables. For C and C++, this does not seem to be an issue since new variables can either be declared at any position in the code (C99/C++) or at the beginning of a new block (C89), which could be opened as part of an `entry` construct's code fragment. However, this approach would require to close the block at the end of the function body, which cannot be accomplished using a simple `exit` construct as this is also applied to intermittent `return` statements. In addition, Fortran requires the declarations of local variables to precede the first executable statement. It therefore seems reasonable to provide a separate `decl` construct to specify local variable declarations:

```
decl file="*" routine="#" code="static int count = 0;"
```

Depending on the purpose of the instrumentation, initializing such a variable with the result of a function call should be avoided, since this would be executed *before* any code fragment specified via an `entry` construct.

4.4 Inclusion of Header Files

Inserting calls to a performance-measurement or debugging API into the source code of an application typically also requires including one or more header files defining the corresponding function prototypes. Fortunately, the TAU instrumentor already provides a mechanism which can be exploited to accomplish this task: using a special `file` rule, some arbitrary code fragment can be inserted at a particular line in the specified source file. For example, the specification line

```
file="*" line=1 code="#include <stdio.h>"
```

can be used to include the header file "stdio.h" at the top of every processed source file.

4.5 Aborting the Application

Other interesting locations where the insertion of, e.g., clean-up code might be useful are calls to the `exit()` or `abort()` functions in C/C++ or the occurrences of the `stop` keyword in Fortran. For this purpose, the `abort` construct is provided:

```
abort file="*" routine="#" code="printf(\"Abort\\n\");"
```

As already described in Sec. 4.2 in the context of the `exit` construct, the keyword substitutions can be used to distinguish different abort locations from each other.

4.6 Initialization

Finally, a tool library might need to be initialized before any other API call is executed. For C and C++ this could be accomplished by providing an `entry` rule restricted to the function `main()`, however, for Fortran the name of a program can be arbitrary. Therefore, a separate `init` construct is necessary:

```
init file="*" code="init_api();"
```

This construct does not need a `routine` part, as it implicitly applies to `main()` in C/C++ or the main program routine in Fortran, respectively.

As a tool library might want to parse the command line arguments given to the instrumented application, e.g., to configure a measurement run, two special keyword substitutions have been implemented for the `init` construct, although for C and C++ only. In this case, the names of the first and second parameter of `main()` are substituted for the `@ARGC@` and `@ARGV@` keywords, respectively. If `main()` has been defined without arguments, the names of two artificially created local variables are inserted, providing the values "1" and "unknown".

4.7 Restricting Rules to a Language

Although it is possible to create separate instrumentation specification files for each supported programming language, we believe that it is more convenient to keep everything together in a single file. All of the aforementioned constructs therefore support an optional `lang="..."` part taking a comma-separated list of language names, restricting the corresponding specification clause to only those languages.

4.8 Language Peculiarities: Line-Length Limit, Templates and Exceptions

Due to the keyword substitutions performed at instrumentation time, the actual lengths of the code fragments to be inserted are not known in advance. This poses a problem in the context of instrumenting Fortran codes, since the maximum length of an individual source line is restricted by the language standard. It is therefore necessary to preprocess the code snippets before inserting them into the output file and eventually introduce additional line breaks and continuation marks, taking into account whether fixed-format or free-format style is used.

Another challenging language feature are C++ templates. At instrumentation time, the `@ROUTINE@` keyword substitution can only provide the generic template prototype, but not the concrete instantiation. If it is a class or member-function template, however, this information can be queried using the run-time type information system (RTTI). As this constraint can be verified at instrumentation time, an additional keyword substitution for `@RTTI@` can be performed, which either expands to `typeid(*this).name()` in case of a class or member-function template or the generic template prototype (i.e., the same value as for `@ROUTINE@`) otherwise. However, the value of the `typeid` expression is compiler-dependent and might be a linker decorated name, which has to be taken into account when using this feature.

Finally, a source-code instrumentor can handle C++ exceptions only to a certain extend, since this is a highly dynamic language feature. Although `throw` statements could be instrumented similar to `return` statements, they do not necessarily leave only the current routine, but all routines up to the next matching `catch` block. However, tools can leverage destructors of local objects [11] to get a correct sequence of exit events.

5 Evaluation

To evaluate whether the proposed specification clauses presented in the previous section are already sufficient to satisfy the needs of current tools to perform a simple per-routine entry/exit instrumentation, we have implemented them in our prototype based on the TAU instrumentor, except for the language-specific features described in Section 4.8. Afterwards, we have developed a set of specification files for a number of performance-analysis toolsets using their manual instrumentation API and verified their correct mode of operation by applying the instrumentor to various test codes.

Our first target was the Scalasca toolset [12]. As the documented user instrumentation API is basically a set of convenience C preprocessor macros heavily using the predefined names `__FILE__` and `__LINE__`, we had to use the lower-level routines these macros are build upon. For all three supported languages, a header file defining the API had to be included. In addition, instrumenting C++ code only required a single `entry` construct due to the availability of a measurement class employing the aforementioned "local object" technique. By contrast, instrumenting C code required the `entry` and `exit` constructs, as shown in the following self-contained example:

```
file="*" line=1 code="#include <epik_user.h>"
entry code="EPIK_User_start(\"@ROUTINE@\", \"@FILE@\", @BEGIN_LINE@);"
exit code="EPIK_User_end(\"@ROUTINE@\", \"@FILE@\", @END_LINE@);"
```

For Fortran, an additional local variable needed to be declared, i.e., the `decl` construct had to be used as well. In all three cases, the `@ROUTINE@`, `@FILE@` and `@LINE@` keyword substitutions were sufficient to fully exploit the current functionality of the provided instrumentation API.

As a second example, we investigated the VampirTrace performance measurement system [13]. Providing an instrumentation API very similar to Scalasca, it was straightforward to come up with a specification file using the same constructs. For both toolsets, the instrumentor could in fact provide more details about source-code locations than necessary, indicating potential for extending the tool APIs to collect even more expressive information.

A far more challenging problem was to clone the TAU instrumentation originally performed by the instrumentor using the generic specifications. It turned out that all of the constructs described in Sec. 4 are needed to fulfill this task. However, two minor issues still remained were the original TAU instrumentor behaved differently.

First, TAU supports so-called *profile groups* as a mechanism to further classify sets of functions. The default behavior of the instrumentor for C and C++ codes is to add the program's main function to the group `TAU_DEFAULT` and all the other functions to the group `TAU_USER`. This behavior could be partially emulated by specifying a separate `entry` rule restricted to the `main()` function, however, there is currently no way of restricting a clause to every function except `main()`. This issue could potentially be solved by supporting full regular expressions in the `routine` part of the specification rules.

The second issue will show up once we have fully implemented template support in our prototype as proposed in Sec. 4.8 because the `@RTTI@` keyword substitution has slightly different semantics than what the TAU measurement system currently assumes. Here, a minor change to the measurement system API would be required, however, this could be implemented as an extension not to break backward compatibility.

6 Conclusion

In this paper, we have investigated which basic constructs are required to specify a user-defined function entry/exit instrumentation in a generic way. We identified six different constructs as the "building blocks" that can be applied to individual files, routines or programming languages, as well as a set of keyword substitutions to take advantage of instrumentor knowledge at instrumentation time. We then evaluated the applicability of the proposed constructs by defining appropriate specification files for three different performance-analysis toolsets and showed that this small set of constructs can already fulfill almost all the needs of a number of todays' tools with respect to routine enter/exit instrumentation. Our prototypical implementation supporting all described constructs except for the language-specific features described in Section 4.8 is available as part of the PDT distribution.

As part of our future work, we plan to first address the open language-specific issues mentioned in Section 4.8. In addition, we will investigate how the configurability can be extended beyond the current enter/exit instrumentation, e.g., to support instrumenting `throw` statements as well as `try` and `catch` blocks, loops, or specially marked program

phases such as OpenMP regions. As a result, the configurable source-code instrumentor component described in this paper should ultimately be able to replace the existing special-purpose instrumentators currently used by various toolsets.

Acknowledgments. This work has been supported by the U.S. Department of Energy, Office of Science under Grants No. DE-FG02-07ER25826 and DE-FG02-05ER25680 and by the Helmholtz Association of German Research Centers under Grants No. VH-NG-118 and VH-VI-228.

References

1. Shende, S.S., Malony, A.D.: The TAU parallel performance system. International Journal of High Performance Computing Applications 20(2), 287–331 (Summer 2006)
2. MPI Forum: MPI – A Message-Passing Interface Standard, Version 2.1. ch.14 (June 2008)
3. MPI PERUSE: An MPI extension for revealing unexposed implementation information (May 2006), `http://www.mpi-peruse.org`
4. Leko, A., Bonachea, D., Su, H.H., George, A.D.: GASP: A performance analysis tool interface for global address space programming models, specification version 1.5. Technical Report LBNL-61606, Lawrence Berkeley National Lab (September 2006)
5. Free Software Foundation: GCC 4.3.2 manual – options for code generation conventions (2008), `http://gcc.gnu.org/onlinedocs/gcc-4.3.2/gcc/Code-Gen-Options.html`
6. Buck, B., Hollingsworth, J.K.: An API for runtime code patching. Journal of High Performance Computing Applications 14(4), 317–329 (2000)
7. De Rose, L., Hoover Jr., T., Hollingsworth, J.K.: The dynamic probe class library – an infrastructure for developing instrumentation for performance tools. In: Proc. 15th International Parallel & Distributed Processing Symposium (IPDPS 2001), Washington, DC. IEEE Computer Society, Los Alamitos (2001)
8. Mohr, B., Malony, A.D., Shende, S., Wolf, F.: Design and prototype of a performance tool interface for OpenMP. The Journal of Supercomputing 23, 105–128 (2002)
9. Schordan, M., Quinlan, D.: A source-to-source architecture for user-defined optimizations. In: Böszörményi, L., Schojer, P. (eds.) JMLC 2003. LNCS, vol. 2789, pp. 214–223. Springer, Heidelberg (2003)
10. Lindlan, K.A., Cuny, J., Malony, A.D., Shende, S., Mohr, B., Rivenburgh, R.: A tool framework for static and dynamic analysis of object-oriented software with templates. In: Proc. SC 2000: High Performance Networking and Computing Conference (November 2000)
11. Meyers, S.: More Effective C++. Addison-Wesley, Reading (1996) (Item 9)
12. Geimer, M., Wolf, F., Wylie, B.J.N., Mohr, B.: Scalable paralle trace-based performance analysis. In: Mohr, B., Träff, J.L., Worringen, J., Dongarra, J. (eds.) PVM/MPI 2006. LNCS, vol. 4192, pp. 303–312. Springer, Heidelberg (2006)
13. Knüpfer, A., Brunst, H., Doleschal, J., Jurenz, M., Lieber, M., Mickler, H., Müller, M.S., Nagel, W.E.: The Vampir performance analysis tool set. In: Resch, M., Keller, R., Himmler, V., Krammer, B., Schulz, A. (eds.) Tools for High Performance Computing, pp. 139–155. Springer, Heidelberg (2008)

Dynamic VO Establishment in Distributed Heterogeneous Business Environments

Bartosz Kryza[1], Lukasz Dutka[1], Renata Slota[2], and Jacek Kitowski[1,2]

[1] Academic Computer Centre CYFRONET-AGH, Cracow, Poland
[2] Institute of Computer Science, AGH-UST, Cracow, Poland
{bkryza,dutka,rena,kito}@agh.edu.pl

Abstract. As modern SOA and Grid infrastructures are being moved from academic and research environments to more challenging business and commercial applications, such issue as control of resource sharing become of crucial importance. In order to manage and share resources within distributed environments the idea of Virtual Organizations (VO) emerged, which enables sharing only subsets of resources among partners of such a VO within potentially larger settings. This paper describes the Framework for Intelligent Virtual Organizations (FiVO), focusing on its functionality of enforcing security (Authentication and Authorization) in dynamically deployed Virtual Organizations. The paper presents the overall architecture of the framework along with different security settings which FiVO can support within one Virtual Organization.

1 Introduction

Modern applications of Service Oriented Architectures or Grid computing are oriented on allowing distinct heterogenous organizations to share their resources in order to pursue some goal through advanced collaboration schemes supported by their IT infrastructures. The Grid idea introduced the concept of Virtual Organization, which abstracts the notion of organization into a virtual environment based on distributed computing infrastructures of organizations that want to collaborate. The idea of Virtual Organization allows these partners to define rules of cooperation in terms of authorization policies or SLA parameters which specify how their resources can and should be shared. The major problem that this idea is facing currently is the problem of very high administrative burden which is required to create and maintain a VOusing available tools. Additionaly no standard currently exists which would allow to actually specify the rules of cooperation by these organizations in a unified manner which then could be used to create the VO by means of configuring proper middleware as well as monitoring the VO operation in order to ensure that the rules which were agreed upon are properly respected. These issues are especially important in case of security focused dynamic environments where creation of VO cannot be delegated to regular system administrators. In order to support creation and management of such dynamic Virtual Organizations, the middleware must provide support

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 709–718, 2009.

for several issues, including resource sharing policy definition and enforcement, resource discovery and usage limited according to the VO policy and others.

In this paper we present our framework, called FiVO (Framework for intelligent Virtual Organization) that supports creation and management of dynamic Virtual Organizations with special focus on dynamic VO creation through contract negotiation and authorization of access to resources. The main feature of FiVO is the contract negotiation and management component, which enables coordinated establishment of agreement among partners who want to create a new Virtual Organization. The contract provides the information necessary for configuration of the VO in the system and allows for specification of both functional and non-functional parameters of the envisioned VO collaborations. Many problems related to ad-hoc creation of a VO are mostly related to heterogeneity of resources shared by VO members. Not only computer equipment is different, but also data formats, service descriptions, knowledge repositories. These issues require a method of mediation between VO members which is necessary to provide connectivity and make collaboration possible and efficient. With respect to VO deployment and contract enforcement we currently focus on security, i.e. authentication and authorization of requests to VO resources based on the contract statements. Other aspects will be developed as the framework evolves further.

2 FiVO Architecture

The architecture of FiVO is based on several goals. The first aim is to provide a unified semantic interface at the service level for discovery and management of all aspects relating to a Virtual Organization, including its members, agreements, goals, resources, data and services. This functionality is achieved through unification of metadata used to annotate these resources with the use of ontologies. FiVO is oriented strictly towards Service Oriented Architectures and Grid computing, thus assuming certain requirements on the infrastructures of organizations willing to participate in Virtual Organizations. These requirements however comply with the current trends in existing and emerging standards for information systems integration. FiVO supports both static and dynamic creation and deployment of Virtual Organizations aiming to pursue some goal, e.g. an emerging market opportunity. In order to enable using this solution in legacy environments the framework supports legacy information systems by providing adapters for existing middleware components. For instance MyProxy [1] and VOMS (Virtual Organization Membership Service) [2] adapters allow to configure these components after successful VO negotiation in order to support GSI based authentication according to the rules defined in the negotiated contract. Such features are of major importance, since legacy information systems and middleware components provide functionality crucial to VO management, especially at the low level and should be reused it in order to make integration between partners of a VO more feasible.

An important aspect of Virtual Organization lifecycle management supported by our framework is contract negotiation, which is a process involving all

partners who want to participate in the emerging Virtual Organization. The contract itself is a set of statements which define terms (statements) related to the goal of the Virtual Organization on which all partners agree. Current FiVO provides framework for manual contract negotiation through a collaborative environment (either Protege plug-in or web-based). Negotiated contract, which is stored in the form of an ontology, is subsequently used to automatically configure the underlying security layer and QoS monitoring components in order to ensure contract enforcement during Virtual Organization operation. This enables organizations to state their commitments and requirements in an abstract way, while minimizing the burden on administrators who otherwise would have to configure all these parameters manually.

The essential contribution of FiVO to existing Virtual Organization management systems is in provision of a unified semantic interface for discovery and management of all aspects of a Virtual Organization (including its members, agreements, resources, goals, SLA's), allowing for dynamic inception through contract negotiation and automatic configuration of underlying middleware layer for contract enforcement during Virtual Organization operation. Figure 1 presents a sample deployment of the FiVO framework in some distributed environment. Four organizations can be seen, sharing their resources within the VO-1. FiVO component is deployed within each organization and is responsible

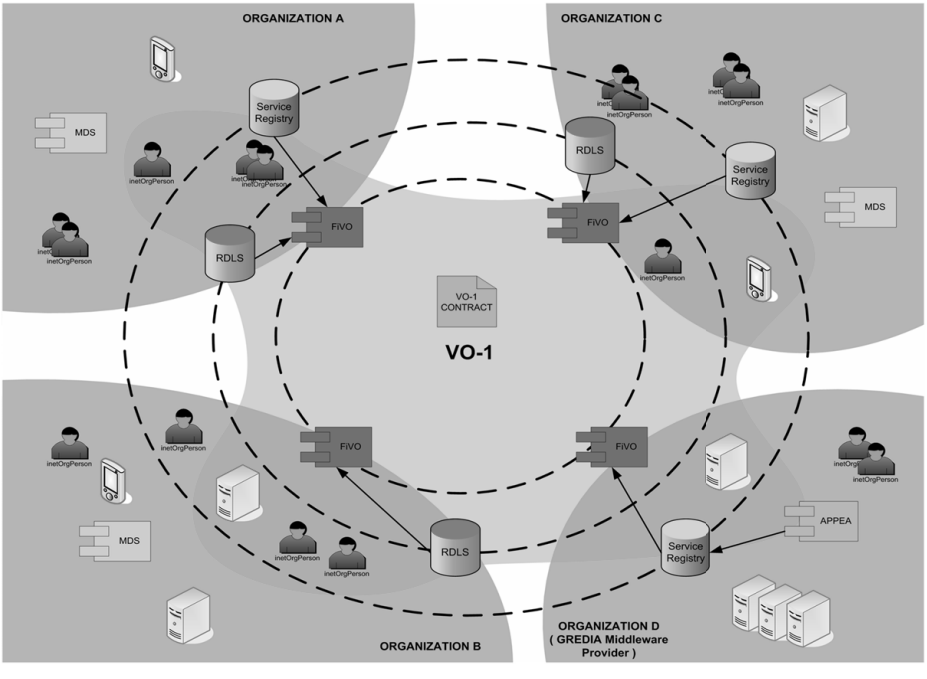


Fig. 1. Vision of the FiVO framework in a distributed environment

for managing this organizations assets, assuming they are described semantically. During the contract negotiation phase, the FiVO components provide for the negotiators view on the resources of the participating organizations which can be used to create proper statements in the contract. After the contract is successfuly negotiated the semantic description of these resources along with the statements from the contract are used to configure proper middleware components. These descriptions can include such aspects of organization as its structure and business logic described in proper ontology as well as hardware, data and service resources available and provided for sharing with other organizations.

3 Contract Negotiation

In order to support the contract negotiation functionality we have defined a formal contract negotiation model and implemented it using Web Ontology Language (OWL) [3]. The vocabulary for the process of contract negotiation is defined by a set of ontologies providing common high-level terminology which can be further instantiated by proper domain level ontologies describing the domain and resources of organization participating within the negotiations. The negotiation process is controlled by a special Graphical User Interface developed for this purpose as a Protege [7] plug-in, see Figures 2 and 3. The formal model gives means to implement a negotiation framework which allows parties to define the rules of cooperation within a given VO. The formal model, as defined

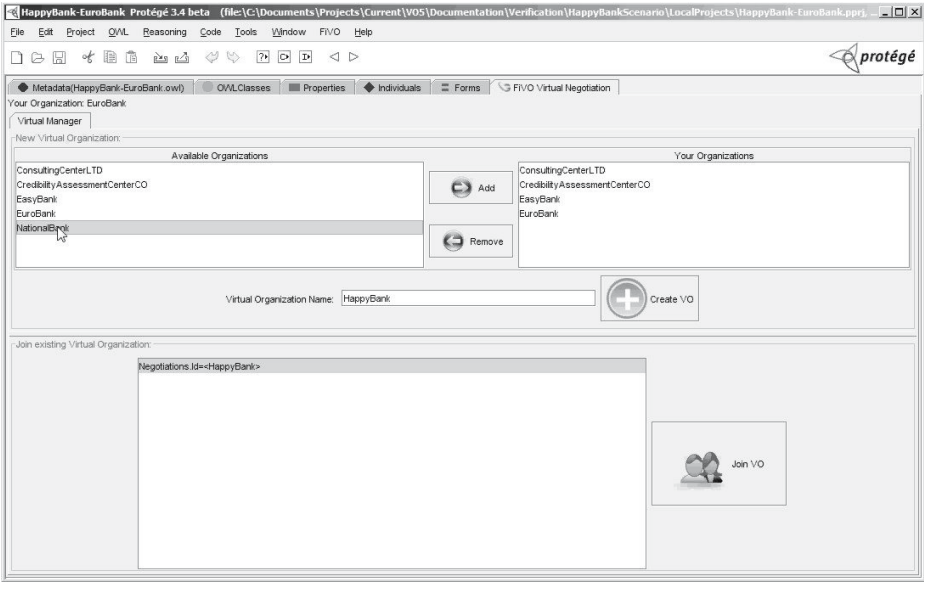


Fig. 2. Joining VO contract negotiations

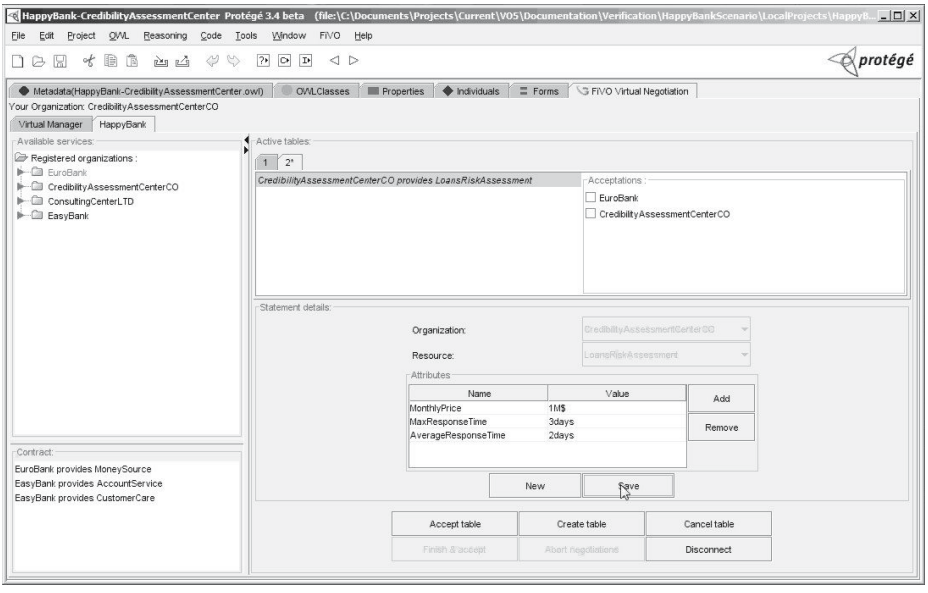


Fig. 3. Negotiating a statement within a negotiation table

in our previous work [6], defines the contract negotiation process in terms of messages exchanged between agents who control resources representing assets of their respective organizations. The negotiation process allows the negotiation of parts of the contract within negotiation tables, thus dividing the complete negotiation process into subnegotiations where only the parties which are directly concerned with the resources that particular statements address need to discuss. In the case of dynamic VO change, for instance when new resources should be added to the VO, contract amendment can be achieved by performing a new contract negotiation phase with the current contract as a starting point. In case the change in resources provided by the partners to the VO is not planned but is caused for instance by a service failure, the contract allows to detect that one of the partners does not fulfill his obligation to the VO and proper steps can be taken to deal with the problem.

4 Semantically Supported VO Security

The main motivation of supporting various security deployment configurations is to enable integration of heterogeneous IT infrastructures within a single VO. After the contract between the parties is negotiated, its rules are used to generate proper entries in MyProxy and VOMS components (i.e. set of roles and their mappings), and from then on FiVO login method uses these components in order to generate a X.509 proxy certificate for a given user. This credential can be then used anywhere within the Virtual Organization, e.g. presented to a Globus

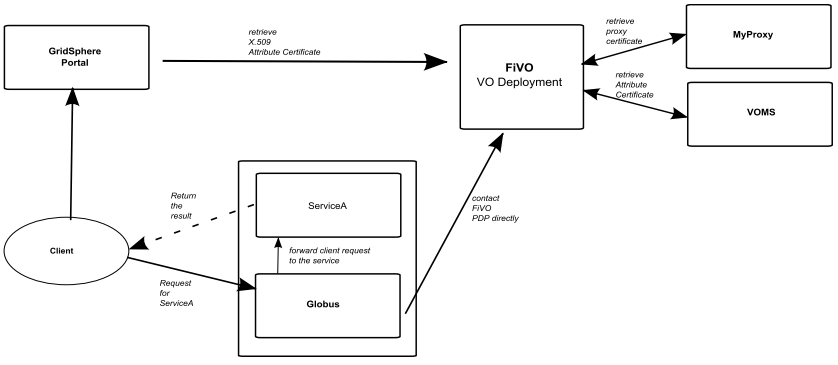


Fig. 4. Globus container and FiVO authentication and authorization

container or Apache web server. These containers will then contact FiVO PDP (Policy Decision Point) providing the role extracted from the proxy certificate presented by the user and receive an access decision.

Currently, FiVO supports 3 basic security deployments with Grid or SOA based environments, as depicted in Figures 4, 5 and 6. In the first mode (Figure 4), when a Globus based WSRF service running in a Globus container is called by a remote client, proper Globus interceptors retrieve the Attribute Certificate from the users credential and send them to FiVO Authorization Service in order to get a AccessGranted/AccessDenied decision. The next mode (Figure 5) allows an Apache based web service (or any resource served by Apache server for that matter), a special Apache module - mod_authz_fivo - retrieves the credentials

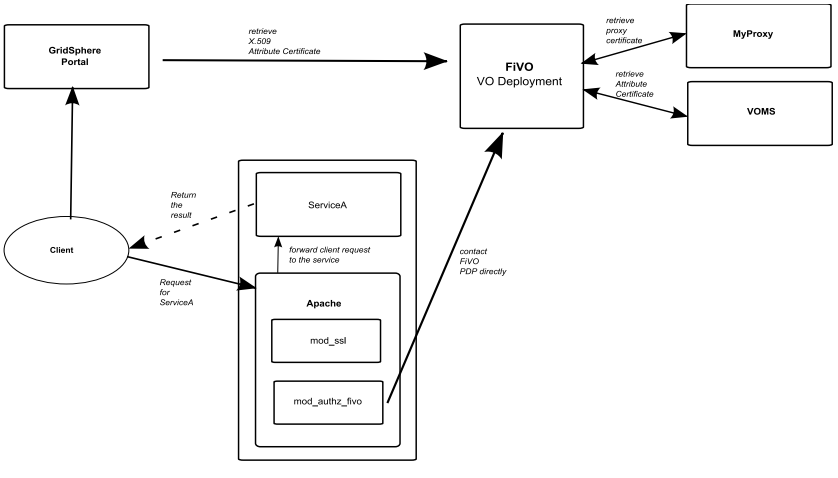


Fig. 5. Apache with FiVO authentication and authorization

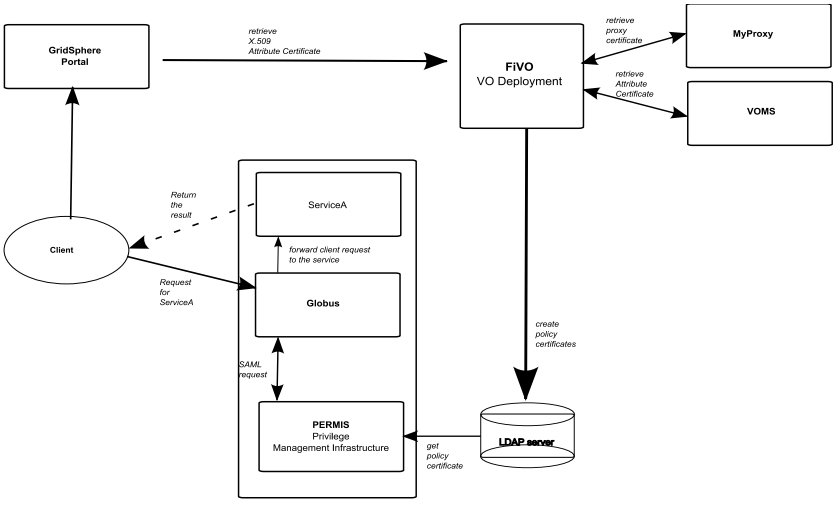


Fig. 6. Globus with FiVO authentication and PERMIS authorization

from the users certificate and contacts FiVO to get the authorization decision. Then it either continues the request processing or returns to the user HTTP Access Forbidden error. In the case of PERMIS (Figure 6), the situation is more complex. PERMIS [8] provides a standalone authorization service (PERMIS Policy Decision Point), however it depends on the LDAP service which contains proper information in terms of Attribute Certificates and Policies. FiVO works with PERMIS by feeding an LDAP service with proper policies, generated automatically from policies which are part of the VO contract. Currently there is an ongoing effort to integrate VOMS Attribute Certificates with PERMIS PDP in the framework of EU-IST VPMan project, so full integration of this approach will be possible as this project published their results. The authorization policies are defined as part of the ontology defining the contract between the partners of a VO, some sample policies are explained below. The first example presents a authorization rule allowing users with role BankManager to access a banking service. The rule assumes that the user has a role stored in a X509 credential, the action as reported by the Policy Enforcement Point (e.g. mod_authz_fivo Apache module) is Execute from a proper FiVO security ontology and the object is identified by the PEP as individual individual from the ontology describing the VO resources.

```
<so:AccessGrant rdf:ID="AccessGrantPOPSOBankServices">
 <so:hasAction
   rdf:resource=".../SecurityOntology.owl#Execute"/>
 <so:hasObject
   rdf:resource=".../FiVO/EasyLoan#POPSOBankService1"/>
 <so:hasSubject>
```

```
   <so:X509AuthorizationSubject
         rdf:ID="X509AuthorizationSubject_BankManagers">
    <so:hasRole>
     <j.2:Role rdf:about=".../FiVO/EasyLoan#BankManager"/>
    </so:hasRole>
   </so:X509AuthorizationSubject>
  </so:hasSubject>
 </so:AccessGrant>
```

The second example presents an authorization rule allowing users with the role Editor to remove files from the system which have certain attributes, in this case location is equal to Iraq. The PEP is the data location service which checks with FiVO whether the user with role Editor can remove the files which have certain metadata attributes.

```
<so:AccessGrant rdf:ID="AccessGrant1">
 <rdfs:comment
       rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
   Grant access for Editors when file has
   attribute location with value Iraq.
 </rdfs:comment>
 <so:hasSubject>
  <so:X509AuthorizationSubject
       rdf:ID="X509AuthorizationSubject_Editors">
   <so:hasRole rdf:resource="#Editors"/>
  </so:X509AuthorizationSubject>
  </so:hasSubject>
  <so:hasAction rdf:resource="#Removes"/>
  <so:hasObject>
   <so:RDLSFileMetadata rdf:ID="RDLSFileMetadata1">
    <so:hasAttributeValue
      rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
       location:Iraq</so:hasAttributeValue>
    </so:RDLSFileMetadata>
 </so:hasObject>
</so:AccessGrant>
```

As can be seen, the use of ontological definition of policies allows for very flexible integration of various types of subjects.

5 Related Work

The enforcement of the contractual agreement is assumed to be handled by the underlying middleware, i.e. including both the security and SLA layers. Security aspects, especially with relation to dynamic Virtual Organization mainly

should revolve around handling heterogeneity of middleware available in such distributed environments. Currently several solutions exist for both authentication and authorization issues which are essential to allow disparate organizations to cooperate together by their resources within interconnected IT infrastructures. The most common authentication systems include X.509 Public Key Infrastructure [9], Kerberos [10] and Shibboleth [11]. Additionally to authentication, the most popular authorization infrastructures for the SOA and Grid based systems include Virtual Organization Membership Service [2], Community Authorization Service [12], Akenti [13] and PERMIS [14]. In [15] the authors describe integration of GSI and Shibboleth based security infrastructures within a VO based on an abstraction approach. In [16] authors present web-Pilarcos J2EE based agent framework for managing contract based Virtual Organizations. The contract itself is an object (J2EE EntityBean) and can be in several states, such as In-negotiation, Terminated etc. The proposed solution is not based on ontologies, and the metadata reasoning is mentioned only briefly. Unfortunately, none of these tools allows suffciently rich resource description framework that would allow to embrace the heterogeneity of infrastructures and environments found in Virtual Organizations, and more importantly their interoperability is very limited.

6 Conclusions and Future Work

In this paper we presented the motivation and architecture of the FiVO (Framework for Intelligent Virtual Organizations) which enables contract negotiation and management for Virtual Organizations which can be applied in heterogenous IT infrastructures such as demanding business settings. We believe that such functionality will foster the adoption of Virtual Organizations in commercial applications by simplifying the process of Virtual Organization inception and management of agreements specifying how the resources of each participating organizations should be shared among partners of a VO. The future work will be focused on provision of additional adapters allowing integration with other popular existing middleware solutions, such as configuring of monitoring components or setting up legacy software for use in a VO, and implementation of automatic contract negotiation agents based on the FiVO framework allowing fully autonomic VO deployment in case of emerging need for such coolaboration.

Acknowledgments

This research has performed done within the framework of EU IST FP6-34363 Gredia project. The authors would like to thank the whole project consortium for remarks and feedback. AGH University of Science and Technology grant nr 11.11.120.777 is also acknowledged.

References

1. Basney, J., Humphrey, M., Welch, V.: The MyProxy online credential repository. Softw., Pract. Exper. 35(9), 801–816 (2005)
2. Alfieri, R., Cecchini, R., Ciaschini, V., dell'Agnello, L., Frohner, A., Lrentey, K., Spataro, F.: From gridmap-file to VOMS: managing authorization in a Grid environment. Future Generation Comp. Syst. 21(4), 549–558 (2005)
3. Antoniou, G., Van Harmelen, F.: Ontology Language: OWL. In: Handbook on Ontologies: International Handbook on Information Systems, pp. 67–92 (2004)
4. Kryza, B., Majewska, M., Pieczykolan, J., Slota, R., Kitowski, J.: Grid organizational memory - provision of a high-level grid abstraction layer supported by ontology alignment. Future Generation Comp. Syst., Grid Computing: Theory, methods and Applications 23(3), 348–358 (2007)
5. Kryza, B., Pieczykolan, J., Kitowski, J.: Grid organizational memory: A versatile solution for ontology management in the grid. In: Proc. of 2nd Intl. Conf. on e-Science and Grid Computing, Amsterdam, Netherlands, December 4-6, 2006. IEEE Computer Society Press, Los Alamitos (2006)
6. Zuzek, M., Talik, M., Swierczynski, T., Wisniewski, C., Kryza, B., Dutka, L., Kitowski, J.: Formal Model for Contract Negotiation in Knowledge-Based Virtual Organizations. In: Bubak, M., et al. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 409–418. Springer, Heidelberg (2008)
7. Knublauch, H., Fergerson, R.W., Noy, N.F., Musen, M.A.: The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 229–243. Springer, Heidelberg (2004)
8. Chadwick, D.W., Zhao, G., Otenko, S., Laborde, R., Su, L., Nguyen, T.: PERMIS: a modular authorization infrastructure. Concurrency and Computation: Practice and Experience 20(11), 1341–1357 (2008)
9. Landrock, P.: Public Key Infrastructure. In: van Tilborg, H. (ed.) Encyclopedia of Cryptography and Security. Springer, Heidelberg (2005)
10. Steiner, J.G., Neuman, C.N., Schiller, J.I.: Kerberos: An Authentication Service for Open Network Systems. In: Proc. of the Winter 1988 Usenix Conf., pp. 191–202 (1988)
11. Sinnot, R.O., Jiang, J., Watt, J.P., Ajayi, O.: Shibboleth-based Access to and Usage of Grid Resources. In: GRID, pp. 136–143. IEEE, Los Alamitos (2006)
12. Pearlman, L., Welch, V., Foster, I., Kesselman, C., Tuecke, S.: A community authorization service for group collaboration. In: Proc. Of Third IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY 2002), pp. 50–59 (2002)
13. Johnston, W., Mudumbai, S., Thompson, M.: Authorization and Attribute Certificates for Widely Distributed Access Control. In: Proc. of IEEE 7th Int Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE), pp. 340–345 (1998)
14. Chadwick, D., Otenko, A., Ball, E.: Role-based access control with x.509 attribute certificates. IEEE Internet Computing 7(2), 62–69 (2003)
15. Kirchler, W., Schiffers, M., Kranzlmuller, D.: Harmonizing the Management of Virtual Organizations Despite Heterogeneous Grid Middleware: Assessment of Two Different Approaches. In: Proc. Cracow Grid Workshop, Krakow, Poland, October 2008 (March 2009) (in print)
16. Metso, J., Kutvonen, L.: Managing virtual organizations with contracts. In: Workshop on Contract Architectures and Languages, CoALa 2005 (2005)

Interactive Control over a Programmable Computer Network Using a Multi-touch Surface

Rudolf Strijkers[1,2], Laurence Muller[1], Mihai Cristea[1], Robert Belleman[1], Cees de Laat[1], Peter Sloot[1], and Robert Meijer[1,2]

[1] University of Amsterdam, Amsterdam, The Netherlands
[2] TNO Information and Communication Technology, Groningen, The Netherlands
{l.y.l.muller,m.l.cristea,
r.g.belleman,delaat,p.m.a.sloot}@uva.nl,
{rudolf.strijkers,robert.meijer}@tno.nl

Abstract. This article introduces the Interactive Network concept and describes the design and implementation of the first prototype. In an Interactive Network humans become an integral part of the control system to manage programmable networks and grid networks. The implementation consists of a multi-touch table that allows multiple persons to manage and monitor a programmable network simultaneously. The amount of interactive control of the multi-touch interface is illustrated by the ability to create and manipulate paths, which are either end-to-end, multicast or paths that contain loops. First experiences with the multi-touch table show its potential for collaborative management of large-scale infrastructures.

Keywords: programmable network, network management, multi-touch interfaces.

1 Introduction

To enable network transparent end-user connectivity, routing protocols or application-specific paths need to be configured by the network operator. However, networks are often complex infrastructures capable of delivering the same service (e.g. an end-to-end connection) in countless ways. Where various degrees of Quality of Service (QoS) are required, the configuration of all the individual devices or network management system becomes a complex task. In addition, routing services handle only basic forwarding and the effects of deteriorated or failing links. In cases for which end-to-end routing services do not apply, or for cases in which the service of the network moves outside the QoS window, a network operator has to take measures to implement the required service or QoS.

In the context of programmable networks, active network implementations [17] and TINA [5] provide applications the tools to configure the network elements (NE), but lack the ability of translating a higher order specification into detailed behavior of NEs. Higher order specifications of the network service can have many forms. A setup-file or a domain specific language is one extreme. In this paper the other extreme is explored: a dedicated human-network interface in the form of a multi-touch Graphical User Interface (GUI) that supports (1) a direct interface where actions in the form of gestures

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 719–728, 2009.

automatically translate in manipulations on individual network elements, (2) real-time, direct interaction with the network that graphically represents the results of actions and (3) collaborative control-room applications where multiple persons manage or monitor a programmable network simultaneously.

The Interactive Network concept was demonstrated at the Dutch research exhibition booth at Super Computing 2008 (SC08), Austin, Texas. The network contained servers with multiple High Definition (HD) video streams that could be routed to HD screens. The demo setup allowed multiple visitors simultaneously to create, manipulate and remove video streams by the touch of fingers.

The next section provides a summary of related work, both for the networking part and the GUI. Section 3 presents the concepts and overall architecture. In section 4, we describe Interactive Network prototype features and its implementation. In section 5, preliminary results are presented. Section 6 gives insight in the current issues, pointers to future work and concludes the article.

2 Related Work

TINA is one of the first efforts that applied IT accomplishments into the domain of networking. One subproject, ACTranS [3] investigated distributed transaction processing support for networks, which resulted in an Object Management Group [15] compliant architecture for transactional path and connection setup services. In addition, the ACTranS Tutorial Demo was a showcase for setting up, manipulating and removing ATM paths with a GUI. However, ACTranS targeted only ATM technology and did not support more than end-to-end connectivity. Furthermore, the complexity of the architecture and implementation never led to adoption by industry.

Nowadays, large-scale efforts in optical and hybrid networking such as UCLPv2 [8] and GEANT [1] use service oriented architectures. By exposing the network element functions as web services these networks support application programmable light paths, and also offer GUI for path management. Among other GUIs for resource management are HP Openview, VMware Infrastructure Manager and workflow managers, such as WS-VLAM [20] to control grid services. However, all the GUIs remain desktop-based single-user applications for network resource provisioning.

With the introduction of the camera based multi-touch technique called FTIR that Jeff Han [9] presented in 2005, multi-touch systems became affordable for research projects. In comparison to traditional input devices (mouse, touchpad and touch screen), multi-touch systems allow multiple points of interaction simultaneously. Complex interactions that require multiple steps in single touch input systems can be simplified with gestures using multiple fingertips.

Until now multi-touch systems were mainly used for scientific visualization and entertainment projects. Most of these projects are related to one or more of the following categories: photo and video organization, paint applications, fluid simulation [13] or geographic information systems [7]. Recent projects show new applications in the field of graph visualization. These projects are often based on existing applications [10] and are trying to visualize, often static, social or computer networks in order to find common structures or properties. By using multi-touch it becomes possible to view and navigate

through large data sets. However, most of these applications share one common limitation: the application only allows the user to control the view and the layout of the data set. In other words, it is not possible to modify data structures.

3 Concepts and Architecture

Programmable networks are more flexible than traditional networks and offer advantages when more than best effort end-to-end services are needed. However, the task of managing a programmable network is different from normal IP networks. Other than IP networks, there is not a single best effort end-to-end service, but a collection of programmable components that are loaded into network elements by default or which can be uploaded on demand. Hence, the resources in a programmable network need to be coordinated, either by hand or by programs that automate decision processes, such as default shortest path routing for applications, or application-specific paths at application request.

Figure 1 shows the User Programmable Virtualized Network (UPVN) [14] architectural framework, which defines the primary components that apply to all programmable networks. Network elements (NE) contain loadable software objects called application components (AC) and allow implementation of not foreseen or application-specific behavior. NEs can also provide a set of default ACs. ACs allow computer programs to access their service interfaces through network components (NC), which are components embedded in applications that interface with the network. NCs act as proxies that provide redirection, virtualization or composition of AC service interfaces. The manner in which NCs are exposed to applications is application-specific, but also the implementation of communication middleware depends on the application domain. For example, monitoring interfaces might be asynchronous and event-based, while control interfaces that require direct response will be synchronous.

Interactive management of a UPVN requires (1) visualization of network state and (2) interaction methods that allow manipulation of the software components loaded in the network. The human becomes part of the decision process by utilizing network visualization for sense making and using interaction methods to persist decisions.

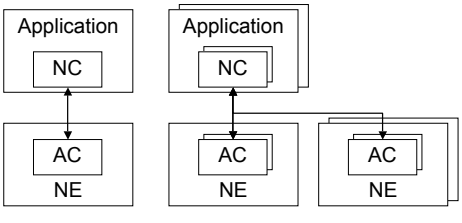


Fig. 1. Relating the UPVN components. Network Components (NCs) are the manifestation of the network in applications. Conversely, Application Components (ACs) manifest as application-specific network services. Implemention choices for NC and AC communication depend on the application-domain.

Figure 2 shows the relationship between the components that make up an interactive network, which is composed of four elementary components.

Network Resource Control. The services provided by individual NEs determine the finest element of detail that can be controlled (Figure 2,1). Services in a NE implement specific adapters or low-level control-loops over NE functions (Figure 2,2).

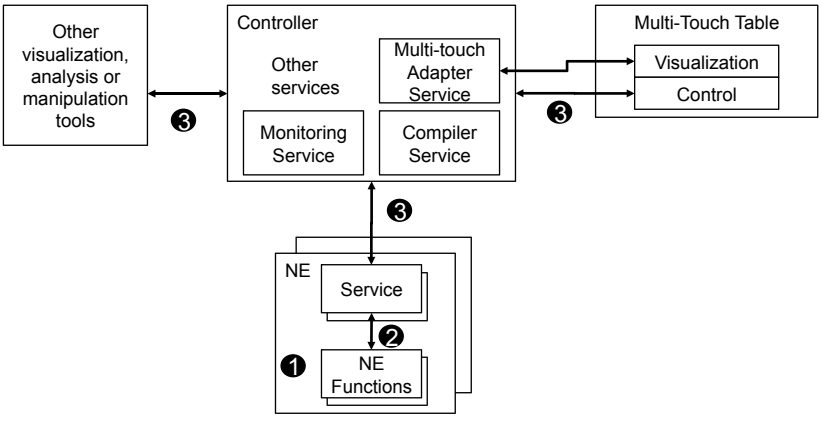


Fig. 2. The architectural components of an Interactive Network: humans interact with the network and have become part of the control loop

A controller application implements services that span over multiple NEs, such as distributed transaction processing or maintaining network topology. The controller application includes the adapter between interaction tools, visualization and human control and for compiling requests into network manipulations.

Middleware. The system has to agree on protocols, services that network elements deliver and their communication mechanisms (Figure 2, 3). Individual services can be discovered and consumed by applications or controllers in the network when implemented by the middleware.

Visualization. The visualization of network state must be in such a manner that it becomes comprehensible to a human and provides useful information for decision support. Visualization of the network spans multiple levels of detail, such as forwarding expressions and rules, routing and inter-domain connections.

Graphical User Interface. The GUI implements interaction with any part of the visualized network. To be suitable for collaborative environments the GUI should support multiple users at the same time that interact with the network. A multi-touch table can provide a suitable environment, because, other than traditional point and click devices, users can gather around the table and use the same GUI.

4 Implementation

We have built an Interactive Network prototype that implements the presented architecture. Figure 3 shows the setup presented in a live demo at the Dutch research exhibition

booth at SC08. The test bed consists of three VMware ESX [18] servers each containing four virtual machines (VM) (Figure 3, 1) running Linux and interconnected by a virtual switch, two commodity Linux servers (Figure 3, 2) and four Mac mini's. A physical gigabit switch connected the VMs and other machines. The Mac mini's were connected through a dedicated path from the booth to our test bed in Amsterdam. There was a separate network (dashed lines in Figure 3) to which the multi-touch table was connected. This allows the programs on the multi-touch table and on the nodes to communicate with the control software, without interfering with the manipulated data streams.

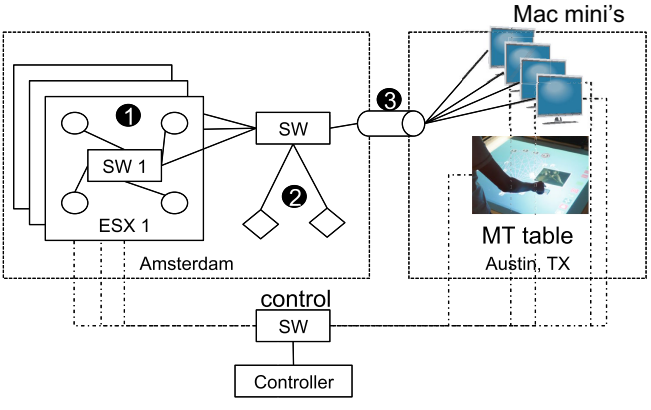


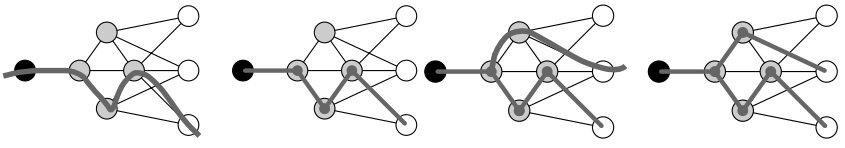
Fig. 3. Hardware setup of the programmable test bed

All the Linux nodes run an AC implemented in Java that provides an interface to local services. Local services implement adapters to among others, Nmap [2] an application to discover neighbouring network elements and Streamline [4], an application that manipulates network traffic in the Linux kernel. A controller application runs on a separate node, which remotely orchestrates behavior of the ACs. The address of this node is known to all nodes part of the network. At boot time, the ACs connect to the controller, which then sends a discovery request to the client to find neighbouring NEs. The controller determines the known topology by combining discovery information of all the ACs, which is then sent to and displayed by the GUI (Figure 4).

Sensing and tracking multi-touch on the table is handled by a separate library, Touchlib [19]. The multi-touch GUI is implemented in Actionscript 3 and runs in the Adobe AIR runtime environment [11]. The implementation supports three different interaction modes: routing mode, local stream manipulation mode and a viewing / monitoring mode.

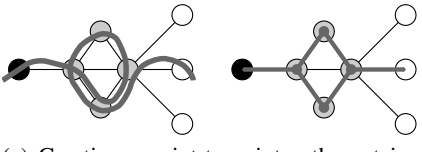
Routing. In routing mode users can define streams from producers to consumers. New streams are defined by dragging a line from node to node, starting with a producer and ending with a consumer (Figure 6(b)). On desktop systems, the most common method is to click the nodes in sequence of the path to create. While it is possible to adopt this method for multi-touch, it also introduces a single-user limitation. Touch and drag can also be used in multiuser environments, such as the multi-touch table.

Fig. 4. Test bed as visualized on the multi-touch table



(a) Creating a point-to-point path.

(b) Extending a point to point path to a multicast path.



(c) Creating a point-to-point path containing a loop.

Fig. 5. Examples showing how to create a path or modify an existing path in the application. The black node is a producer, the grey node a router and the white node a consumer.

At touch, a stream is assigned a unique color, which is also used to identify the traffic in the network. This is achieved by encoding the color id in the IP options field of the IP traffic when provisioning the programmable network. The system supports three kinds of paths:

- Point-to-point drawn from producer to consumer (Figure 5(a)).
- Multicast, drawn by extending an existing point-to-point stream (Figure 5(b)).
- Point-to-point containing a loop. This stream contains a path that crosses a specific node multiple times (Figure 5(c)). We added the ability to create loops to illustrate that some operations, intuitive to a user or application, are not intuitive at all from the perspective of a network. UPVN accommodates such unforeseen application demands, where traditional approaches cannot.

The controller implements a distributed transaction processor to support atomic operations over multiple nodes, such as creating paths. In the case any of the nodes fail while provisioning a path, the whole operation is rolled back and the drawn path is removed from the visualization.

Local stream manipulation. After creating a path in routing mode, users can modify the expressions generated by the path compiler on any node. By double tapping a

node, an overview of currently active streamline expressions is displayed in a zoom window. This way it is still possible to maintain a global overview of the network and still allow other users to interact with other parts of the network.

Streams can be distinguished based on the color assignment at creation. The streamline expressions show the pipeline of filters that are applied to incoming packets before they are sent out. Each filter in the streamline expression represents an application component implemented in a kernel module.

A streamline expression illustrated as a graph can also be modified (Figure 6(c)). Currently, the application only supports adding and removal of sampler filters. Sampler filters allows users to adjust the packet drop rate of a stream. To add a rate limiter to a stream, the user touches a node in the graph and taps the plus sign. This action will send a request to the controller, which will delegate the request to the correct node to update its streamline graph.

Viewing / Monitoring. When multiple users can gather around the multi-touch table, each person will view the display from a different orientation. This becomes an issue in text visualization, which is strongly oriented towards the viewpoint of a user. To avoid this issue, we chose to use as much icons as possible to represent the various node types and GUI elements. An other solution to this issue would be to set default orientations were users can view the multi-touch GUI.

Apart from the network specific interactions, the panel also supports display of monitoring data and manipulations on the network visualization. In viewing mode, for example, the current load on a node can be viewed by touching the icon (Figure 6(a)). A small graph appears that shows information of the current workload and the average workload over the past 5 and 15 minutes.

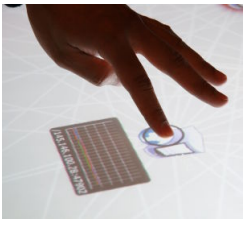
In viewing mode users can also navigate through the network by using the touch of their fingers. Gestures such as dragging and pinching, allow the user to rapidly move the view or zoom into the network. When visualizing a large number of nodes, it becomes difficult to maintain an overview of the network. Therefore, the multi-touch GUI also allows a user to activate automatic layouting. This layout algorithm is provided by the flare library [12].

5 Results

The Interactive Network prototype at SC08 received positive feedback. Visitors were invited to play with the Interactive Network. Multiple users could interact with the system and see the results of their manipulations on the four screens. The direct feedback in terms of video playback proved valuable to help non-network experts to use the system without requiring knowledge of the underlying programmable network.

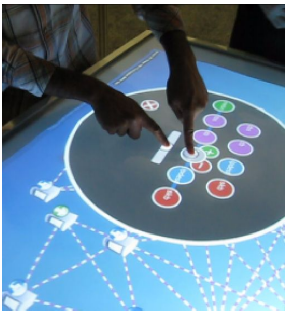
The robustness of the entire test bed was assured by the controller application, which has no a priori knowledge about the nodes that are part of the network. Hence, failure of nodes did not affect the system as a whole. The prototype remained running for four days of testing by visitors without requiring restarts, and was robust against failure or restarting of individual nodes.

The setup also had some limitations. First, due to the multi-touch tracking mechanism environment light had an impact on its accuracy. Second, currently it is not



(a) Monitoring the router node workload.

(b) Creating a new path from a producer to a consumer.



(c) Adjusting a streamline rate modifier.

Fig. 6. Examples of different interaction layers of the application

possible to drag paths and manipulate stream samplers at the same time, because the routing and stream manipulation modes apply to the whole interface. Although in practice this did not pose any problems, for more complex systems it is preferable to support multiple types of interaction at the same time to different parts of the GUI.

6 Conclusion and Future Work

The Interactive Network prototype was designed as a command-and-control environment for programmable networks. By using multi-touch technology we have enabled multiple persons to manage a programmable network simultaneously, without requiring expert knowledge on networks.

Compared to desktop applications, multi-touch devices allow direct interaction with visualization. By controlling the network using multi-touch, management tasks over many network elements can be simplified. When users can have an overview of the system at all times, configuring or programming a collection of network elements is less error prone compared to typing in commands in a terminal for each device separately. However, comparative research is needed to determine the usefulness of multi-touch GUIs compared to traditional point and click devices or collaborative web applications. For example, can the use of multi-touch systems increase the effectivity of collaboration in case of extreme situations, such as sudden calamities in a grid infrastructure or a complex management tasks? In addition, multi-touch enabled automation tools and

software interfaces, such as workflow managers, domain specific (visual) programming languages or schedulers will be needed to support large-scale grid and application management and comparison to the traditional approaches.

The Interactive Network concept can also be applied to other domains, such as Interactive optimization of data centers, include grid services or to interact with computational models that run on grids. The OptiPuter [16] research project, for example, presents a vision of interactive access and control to high-speeds networks, large data-sets and high-performance computing clusters to support e-Science. If the computational and network resources are abstracted and enriched with meta-data, by using NDL [6] for example, multi-touch GUIs can improve the way scientists interact with computational and infrastructural resources to coordinate their experiments and view its results.

Future efforts will include adding semantic information about the network and its resources and include control over more than network services alone. Furthermore, to control grid resources and orchestrate e-Science experiments with multi-touch interfaces will also need research into visualization aspects, such as zooming user interfaces, and intuitive multi-touch gestures.

Acknowledgments

We gratefully acknowledge the help of Willem de Bruijn, Paul Melis, Edwin Steffens and Edward Berbee. This work was supported in part by the European Commission under the project ACGT: Advancing Clinicogenomic Trials on Cancer (FP6-2005-IST-026996) and in the context of the Virtual Laboratory for e-Science (VL-e) project (http://www.vl-e.nl). The VL-e project is supported by a BSIK grant from the Dutch Ministry of Education, Culture and Science (OC&W) and is part of the ICT innovation programme of the Ministry of Economic Affairs (EZ). Part of this project was also funded by GigaPort, Phosphorus and the European Union (EU) under contract number 034115. We also thank the Netherlands Organization for Scientific Research (NWO) for their support and organization of the Dutch research exhibition booth at SC08.

References

[1] GEANT network, `http://www.geant.net`
[2] Network mapper, `http://nmap.org`
[3] ACTS, A transaction processing toolkit for acts: Actrans, final report (1998)
[4] Bos, H., de Bruijn, W., Cristea, M., Nguyen, T., Portokalidis, G.: FFPF: Fairly fast packet filters. In: OSDI (2004)
[5] Chapman, M., Montesi, S.: Overall concepts and principles of tina, Tech. report, TINA Consortium (February 1995)
[6] Dijkstra, F., van der Ham, J.J., Grosso, P., de Laat, C.: A path finding implementation for multi-layer networks. FGCS 25, 142–146 (2009)
[7] Forlines, C., Esenther, A., et al.: Multi-user, multi-display interaction with a single-user, single-display geospatial application. In: Proceedings of the 19th Annual ACM Symposium on User interface Software and Technology. ACM, New York (2006)

[8] Grasa, E., Junyent, G., et al.: UCLPv2: A network virtualization framework built on web services. IEEE Communications Magazine 46 (2008)
[9] Han, J.Y.: Low-cost multi-touch sensing through frustrated total internal reflection. In: Proceedings of the 18th Annual ACM Symposium on User interface Software and Technology. ACM Press, New York (2005)
[10] Heer, J., Boyd, D.: Vizster: Visualizing online social networks. In: Proceedings of the 2005 IEEE Symposium on Information Visualization, Washington, DC, USA (2005)
[11] Adobe Inc., Adobe AIR (2008), http://www.adobe.com/products/air/
[12] UC Berkeley Visualization Lab, Flare: Data visualization for the web (2008), http://flare.prefuse.org/
[13] Han, J.Y.: Multi-touch interaction wall. In: ACM SIGGRAPH 2006 Emerging Technologies. ACM Press, New York (2006)
[14] Meijer, R.J., Strijkers, R.J., Gommans, L., de Laat, C.: User programmable virtualized networks. In: Proceedings of the Second IEEE international Conference on E-Science and Grid Computing (2006)
[15] The Object Management Group (OMG), http://www.omg.org
[16] Smarr, L., Brown, M., de Laat, C.: Editorial: Special section: Optiplanet - the optiputer global collaboratory. FGCS 25, 109–113 (2009)
[17] Tennenhouse, D.L., Wetherall, D.J.: Towards an active network architecture. SIGCOMM Comput. Commun. Rev. 46 (2007)
[18] VMware Inc., VMware, http://www.vmware.com
[19] Wallin, D.: Touchlib: an opensource multi-touch framework (2006), http://www.whitenoiseaudio.com/touchlib/
[20] Wibisono, A., Korkhov, V., et al.: WS-VLAM: Towards a scalable workflow system on the grid. In: Proceedings of the 2nd Workshop on Workflows in Support of Large-Scale Science, Monterey Bay, California, USA (June 2007)

Eye Tracking and Gaze Based Interaction within Immersive Virtual Environments

Adrian Haffegee and Russell Barrow

Advanced Computing and Emerging Technologies Centre,
The School of Systems Engineering, University of Reading, UK
a.haffegee@reading.ac.uk

Abstract. Our eyes are input sensors which provide our brains with streams of visual data. They have evolved to be extremely efficient, and they will constantly dart to-and-fro to rapidly build up a picture of the salient entities in a viewed scene. These actions are almost subconscious. However, they can provide telling signs of how the brain is decoding the visuals, and can indicate emotional responses, prior to the viewer becoming aware of them.

In this paper we discuss a method of tracking a user's eye movements, and use these to calculate their gaze within an immersive virtual environment. We investigate how these gaze patterns can be captured and used to identify viewed virtual objects, and discuss how this can be used as a natural method of interacting with the Virtual Environment. We describe a flexible tool that has been developed to achieve this, and detail initial validating applications that prove the concept.

1 Introduction

Psychological studies often use eye tracking to gather information relating to how a user reacts to particular visuals. Typical uses would be for areas such as marketing aesthetics, the effectiveness of emergency signage, or measuring attention[1,2]. Current methods record the gaze position on a 2D image or video stream, with the captured data being stored for offline analysis.

Virtual Environments (VE) are used to create alternate worlds that users can enter and interact with. These worlds are configurable and controllable, and are well suited for constructing scenes that would be difficult or time consuming to build in the real world e.g., those that are hazardous, dynamic, or too expensive.

By using eye tracking within the VE, it is possible to capture a user's eye movements and analyse how they are observing the scene. Because the composition of the VE is known, these actions can be directly mapped to entities within the scene providing the possibility of automatic analysis of where a user is looking. This could then be used for interaction with the environment itself, with previous work considering sight operated pointing and selecting[3,4]. Although there has been a reasonable research with eye tracking within VEs, there has been little within modern 3D immersive projection technologies. This combination with these systems that provide a more natural interactive experience has

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 729–736, 2009.

wide potential. In multiuser environments, research has been undertaken on the importance of gaze to aid communication amongst remote participants[5,6].

In the next section we will discuss the system configurations that have been used, and why they were chosen. Section 3 will then detail the methods and algorithms used in the implementation. Section 4 will cover the validation and a basic sample application, before Sect. 5 concludes the paper and describes potential future extensions.

2 System Configuration

This research focuses on the problem of enabling a head mounted eyetracker to be used inside an Immersive Virtual Environment (IVE). The intelligent coupling of these two systems makes it possible to calculate where in the virtual scene a user is looking. The main components are the immersive VR system, the eyetracker, and software that binds it together.

Immersive Systems. Immersive projection based technologies such as the CAVE [7] place the user in an environment surrounded by one or more projection screens. The user is free to move within the confines of the system. They are position tracked and the perspective of their viewed images are adjusted according to their relative head position. While more expensive than other immersive technologies such as Head Mounted Displays (HMDs), these systems are less restrictive and provide a more natural view of the environment in addition to the methods of interacting with it. These systems provide the user with a higher degree of *presence* within the VE [8]. This is the degree that they feel that they are a participant within the environment. The greater the feeling is, the more likely they are to instinctively behave and react as though they were within a real situation.

The Mobile Eye. The ASL Mobile Eye eye tracker [9], was used for this project. It is lightweight and glasses mounted, and is well suited for use within IVEs where the user will be free to move around. This is in contrast to freestanding trackers, which require the user to remain relatively still, and at a fixed position from the device. Other head mounted devices could use the approach described, but may need minor modifications depending on the format and structure of their data output.

The Mobile Eye uses Dark Pupil Tracking to calculate gaze position. This uses the pupil position and the reflection from the cornea to determine the eye's direction. Figure 1 shows how we combine the system with a standard head tracker (here an Intersense IS-900), which obtains the user's head position and orientation within the environment. Attached to the glasses are two cameras; one aimed in the direction of the user's vision, and the second capturing an image of their eye reflected by the combiner.

The system is calibrated to provide a gaze position for where the user is looking at any time. This is indicated to the user by overlaying a gaze position marker on top of the output video stream from the scene camera. A sample

Fig. 1. The ASL Mobile Eye device, showing how it is combined with a head tracker (left). Video output from the device, with a crosshair overlay for gaze direction (right).

output from this can be seen in the right hand image of Fig. 1. Here a crosshair is being used to represent the user's line of sight. The (x,y) coordinates of the marker as displayed on the video image can optionally be streamed in an encoded format from the analysis computer's serial port. It is these *Point of Gaze* (PoG) coordinates that we use for calculating our virtual world gaze tracking.

Virtual Environment Application Development. Different methods and tools are available for the creation of IVEs. This research used the VieGen framework[10], which is a set of tools and utilities to aid application development. Entities within the virtual world are represented by members of an extensible family of SceneNodes. These contain the configurable attributes and behaviours of the objects, can be dynamically loaded at runtime, and provide a harness for developers to extend the environment.

3 Calculating and Using the Gaze Vector

This project converts the PoG output from the Mobile Eye into a virtual world gaze vector. This is a vector starting at the user's eye position and heading off in the direction of their line of sight. Within the VE, this vector can be used to indicate potential areas of visual interest, or as advanced methods of controlling the environment. Being glasses mounted, the Mobile Eye's frame of reference is that of the head tracker offset by the distance from the tracker to the eye. This relationship provides a method of converting from the (x,y) PoG coordinate output into the 3D virtual world gaze vector.

Figure 2 shows a breakdown of the modules developed for this research. The left image shows the high level components which have been wrapped into a VieGen Dynamic SceneNode, allowing for rapid development and portability. The Eye Tracking Control Module provides the core functionality from this research, and could easily be extracted for incorporation into any other VR

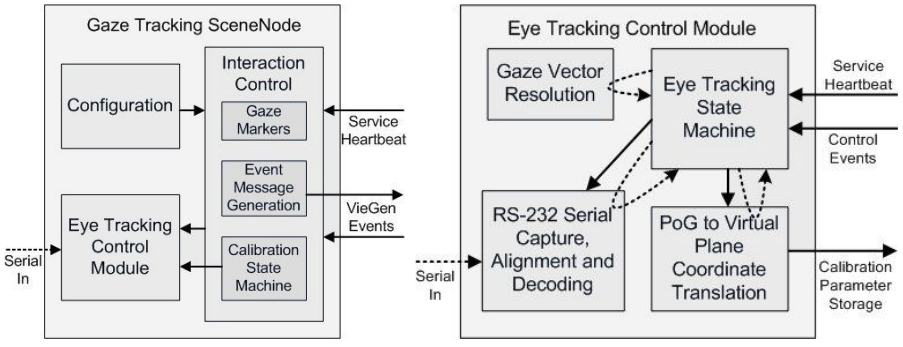


Fig. 2. Gaze Track SceneNode overview (left) & Eye Tracking Control Module (right)

application or framework. The right image shows the breakdown of this core component, containing the RS-232 (serial) stream decoding, coordinate translation, conversion to gaze vector, and the state machine that binds it together. In addition to the control module the SceneNode also contains configuration data and an interaction component responsible for how the node will react with the rest of the VE.

3.1 Eye Tracking Control Module

Serial Capture, Alignment and Decoding. The Mobile Eye streams the encoded tracking information as consecutive 10 byte blocks of serial data. This component locks onto the stream to locate the start of each block, and then decodes the data into a structure which contains the PoG coordinates in the video stream. If the tracker fails to calculate the eye position, (e.g. due to the user blinking or removing the glasses), a status byte within this structure is used to indicate an error condition.

Mapping PoG Coordinates on to a Virtual World Plane. The PoG coordinates can be considered as the (x,y) coordinates on a plane that is a constant distance and perpendicular to the user's head position. A similar plane can be created in virtual space maintaining a fixed position relative to the user's head tracked location. A relationship between the real and virtual gaze positions can be obtained by having the user fixate on a known point on the virtual plane, while reading the PoG coordinates streamed from the Mobile Eye. The software takes several readings for each of these fixation points, and averages the valid ones to minimise errors or inaccuracies. By sampling a number of these relationships across different positions on the gaze plane, a calibration mapping of PoG (x,y) position to virtual plane location can be constructed.

This calibration data is stored within the module, and can be further analysed to determine its nature. It was observed that for the Mobile Eye there was a linear correlation between the PoG coordinates and the virtual plane positions. However it should be noted that different cameras/lenses could deviate from this

and would require subsequent algorithm modifications. Ideally future versions would be able to automatically self-check the calibration data and could prompt for recalibration if required.

From the calibration mapping it is desirable to formulate a method of calculating the virtual gaze plane interception location from any PoG (x,y) position. Assuming that the mapping is linear the gaze plane location can be calculated by comparing the unknown position's PoG value relative to two of the known calibration points. Ideally the chosen points should be sufficiently distant from each other to reduce the effect of errors in the calibration data. The reliability of this calculation can then be further improved by combining the results obtained relative to a number of these calibration point pairs. However small degrees of non-linearity in the mapping would introduce errors the greater the distance to these calibration points.

Obtaining the Gaze Vector. The virtual gaze plane is located at a fixed offset from the user's head, the location of which is known within the VE. By applying the eye offset to the head location the eye position can be found. The gaze vector is a ray starting at the eye and heading through the gaze plane point of interception, and off into the distance. This ray can be applied to the virtual scene to determine the first object that it intercepts. Assuming the object is visible it will be the object being viewed.

3.2 Scene Interaction Control

The gaze tracking functionality is wrapped in a VieGen SceneNode, which provides useful infrastructure for interfacing with the virtual scene. It includes the functionality for calculating the gaze vector, and also a state machine for controlling the calibration process, markers for indicating the direction or position of gaze, and event message handling for informing other scene objects if they are being viewed. Although this section is based on the VieGen infrastructure, the methods and algorithms could be ported to other VE development frameworks.

Calibration. Although the PoG calibration mapping is handled in the eye tracking control module, the process of conducting a calibration run is controlled by the SceneNode. The user is required to sequentially fixate on known points distributed about the virtual gaze plane. A state machine manages this process, displaying the gaze plane in the VE and using a sphere to indicate where the user should be looking. A green sphere is used to indicate reception of valid readings from the mobile eye, and this is changed to red should they switch to invalid. During the sampling process for each sphere position, the gaze plane changes red to inform the user that they need to remain fixated. Once the background returns to grey the sampling will have finished, and the sphere will either move to the next location or will remain in place should it need to be repeated.

Upon completion of all calibration points the system stores the set of calibration data. A simple graphing object within the VE was used to represent the mappings for each of the X and Y coordinates. The left image of Fig. 3 shows

Fig. 3. In-scene visualization of calibration data to validate results (left), & live gaze selection in the virtual gallery, with multiple indicators for the viewed object (right)

a typical output from these graphs for the Mobile Eye. As can be seen, these should display a linear mapping. The provision of graphs that can be viewed within the VE enables in-scene indication of the reliability of the calibration data. This simplifies the process of identifying inaccuracies in either individual calibration points or the complete set.

Gaze Position Markers, and Message Events. A simple way of indicating the gazed object is by adding a marker to the scene at the gaze vector intersection point, however this may not be the best approach. If there is any error from the calibration the marker may be slightly offset from the viewed position. Naturally the users eyes will be drawn to the marker, resulting in a new position for the marker, again slightly offset. This repeats, resulting in a marker that appears to wander the scene. The involuntary eye action can wrongly lead to assumptions about the system stability. This can be resolved by using different markers. One such approach is to use a bounding box around the object being viewed. This provides a more stable indication, but can lose the accuracy as to which part of the object was being viewed. If the scene objects have been named, their textual information can be displayed within the scene, but again this does not convey the exact hit location. An alternative method is not to display the marker in the user's view of the scene, but to store a log of its position for separate display to interested third parties.

Some VEs allow virtual objects to create and consume message events. For VieGen, an event has been defined to indicate that an object is being viewed, and this is forwarded to the first object intersected by the gaze vector. While current reactions to this event are limited, future responses could include moving, animating, or other methods of interaction between the object and the viewer.

Configuration. For this project, the gaze tracking SceneNode can be configured at run time. It uses XML to define the com port to be used for the serial

connection between the analysis computer and the VR system, the offset from the head tracker to the users eye, the position and size of the virtual gaze plane, and the number and locations of calibration points used in the PoG mapping.

4 Validation and Virtual Gallery Application

The practical experiences during the development of this technology have clearly shown that the immersive eye tracking developed has been successful. The basic functionality of the gaze tracking SceneNode enables a gaze marker to be displayed which indicates the location in the environment that the user is viewing. To further prove the features of the technology an eye gaze specific application was developed based around a virtual gallery. This VE consists of a display case containing various artifacts that the user can view. It has been enhanced with in-scene menus to start calibration, add or remove different types of indication markers, and to start/stop the logging to disk of the gaze vector interception positions. The latter of these can be used for subsequent analysis and replay of the user's viewing patterns. The photo on the right of Fig. 3 shows this application in use. The user is able to select the different objects solely by using their eyes, with movements of their head and body having no detrimental effect on the selection process. In this example, a sphere is used as a gazed object marker along with a wireframe bounding box, and these can be seen on the selected rabbit. During the tests, the gaze positions were recorded and these could be replayed within the scene to show the users viewing patterns.

5 Summary and Future Work

While no formal evaluation tests have yet been conducted with this research, the initial results clearly demonstrate its feasibility. Once calibrated the system will reliably follow the users eye position regardless of how they move both in the virtual world and within the confines of the VR system.

However there is still scope for optimization of the algorithms used, particularly in the field of calibration. The relationship between the PoG and the gaze plane should be further studied to determine the nature of this relationship and to investigate if can be represented mathematically rather than as a comparison between calibration points. It would also be useful if the different calibration points could be compared to assess their reliability and accuracy. A further extension could involve modifications to the Mobile Eye analysis software which would allow it to receive scene coordinates from the VR system. It this case the complete system could be calibrated inside the VE, and this would do away with the need for the intermediate gaze plane currently being used.

Additional extensions could also be developed to aid analysis of the captured user data. In addition to walk-throughs of the VE that replay dynamic user head and eye positioning, these could also include hot-spots indicating areas of particular interest. Analysts could explore and navigate these VEs, with superimposed markers demonstrating the viewing behaviour. Indeed, multiple user input files

could be combined to allow more quantitative analysis. These could be filtered as desired, and overlaid on the scene to show key areas of interest.

There is vast potential in the use of this technology, and many researchers may benefit from studying eye behaviour from within a fully controllable environment. Commercially this could include market research or safety analysis, where attracting a user's attention visually is important. Research could include how eye movements are used as communication extensions. By extending the technology as a control or navigation interface, it could also provide a natural method of interaction. This could be especially useful for those with disabilities that preclude them from otherwise controlling or participating within the VE.

References

1. Duchowski, A.T.: Eye Tracking Methodology: Theory and Practice. Springer, New York (2007)
2. Cox, A.L., Cairns, P., Berthouze, N., Jennett, C.: The use of eyetracking for measuring immersion. In: CogSci 2006 Workshop: What have eye movements told us so far, and what is next?, Vancouver, Canada (July 2006)
3. Tanriverdi, V., Jacob, R.J.K.: Interacting with eye movements in virtual environments. In: CHI, pp. 265–272 (2000)
4. Asai, K., Osawa, N., Takahashi, H., Sugimoto, Y.Y., Yamazaki, S., Samejima, M., Tanimae, T.: Eye mark pointer in immersive projection display. In: VR 2000: Proceedings of the IEEE Virtual Reality 2000 Conference, Washington, DC, USA, p. 125. IEEE Computer Society, Los Alamitos (2000)
5. Garau, M., Slater, M., Vinayagamoorthy, V., Brogni, A., Steed, A., Sasse, M.A.: The impact of avatar realism and eye gaze control on perceived quality of communication in a shared immersive virtual environment. In: CHI 2003: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 529–536. ACM, New York (2003)
6. Murray, N., Roberts, D., Steed, A., Sharkey, P., Dickerson, P., Rae, J.: An assessment of eye-gaze potential within immersive virtual environments. ACM Trans. Multimedia Comput. Commun. Appl. 3(4), 1–17 (2007)
7. Cruz-Neira, C., Sandin, D.J., Defanti, T.A., Kenyon, R.V., Hart, J.C.: The CAVE: Audio visual experience automatic virtual environment. Communications of the ACM 35(6), 64–72 (1992)
8. Slater, M., Steed, A., Chrysanthou, Y.: Computer Graphics and Virtual Environments. Addison Wesley, Reading (2002)
9. Applied Science Laboratories: Operation Manual Mobile Eye (January 2007)
10. Haffegee, A.: VieGen: An Accessible Toolset for the Configuration and Control of Virtual Environments. Ph.D thesis, University of Reading (March 2008)

Collaborative and Parallelized Immersive Molecular Docking

Teeroumanee Nadan[1], Adrian Haffegee[1], and Kimberly Watson[2]

[1] Advanced Computing and Emerging Technologies Centre
[2] School of Biology Unit
University of Reading
Reading, RG6 6BX, United Kingdom
t.nadan@reading.ac.uk

Abstract. During docking, protein molecules and other small molecules interact together to form transient macromolecular complexes. Docking is an integral part of structure-based drug design and various docking programs are used for *in-silico* docking. Although these programs have powerful docking algorithms, they have limitations in the three-dimensional visualization of molecules. An immersive environment would bring additional advantages in understanding the molecules being docked. It would enable scientists to fully visualize molecules to be docked, manipulate their structures and manually dock them before sending to new conformations to a docking algorithm. This could greatly reduce docking time and resource consumption. Being an exhaustive process, parallelization of docking is of utmost importance for faster processing. This paper proposes the use of a collaborative and immersive environment for initially hand docking molecules and which then uses powerful algorithms in existing parallelized docking programs to decrease computational docking time and resources.

Keywords: Virtual Reality, molecular visualization, parallelization, collaborative, immersion.

1 Introduction

A receptor is a large protein molecule with a cavity/pocket/active site in which a smaller molecule, called a ligand, fits into and interacts with to form a complex molecule having complex biological functions. This interaction between a receptor and a ligand is known as a docking event. Docking can include interactions of various complexities ranging from less to more complex, such as protein-protein, protein-DNA, protein-metal and protein-ligand interactions. Various *in-silico* docking programs have been developed, each with powerful docking algorithms, to best dock the ligand in a receptor. Due to the complexities involved, the docking process is time consuming. It is therefore important to parallelize docking in order to speed up the process. However, to best reduce docking time, it is helpful to initially hand dock the ligand into the receptor's active site. Experts have

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 737–745, 2009.

insight and can 'see' potential docking possibilities. Thus, a proper visualization medium must be available to enable experts to better understand three dimensional (3D) molecular structures and to provide intelligent guesses for starting the docking process.

This paper proposes the use of a parallelized docking algorithm in an immersive environment to aid the docking process. In an effort to make maximum use of scientists' knowledge, it has been decided to visualize and hand dock molecules in a Cave Automatic Virtual Environment (CAVE) [1]. Thus, instead of performing docking directly (i.e. in the absence of user input), users will be able to first manually dock the molecules and then computationally dock them by using docking programs. The proposed system will make use of robust algorithms from parallelized desktop-based docking programs to generate the docking results. Collaboration will be introduced to allow multiple users to view/manually dock molecules and also to view the docking results. This will allow multiple users to share their knowledge and optimally hand dock the molecules.

2 Molecular Docking Challenges and Related Work

The motivation behind this project is three-fold, namely: visualization, collaboration and parallelization.

2.1 Limited Visualization

In-silico docking programs generally are limited to desktop use, hindering scientists from fully visualizing structures in 3D and limiting the visual information. The user interface for desktop-based computational programs usually uses a mouse, reducing interaction with the molecules being visualized. However, it is crucial to be able to rotate, translate and manipulate molecules in multiple dimensions to emphasise on hidden details. For better docking, it is imperative to have a good visualization of the receptor's and the ligand's three dimensional positions and interaction with them. To overcome these limitations, it is important to provide an immersive environment which can bring all or most of the advantages of such an environment to the user. This can best be achieved in a Virtual Reality (VR) environment, enabling a better visualization of the molecules, and hence a better understanding of the receptor-ligand interactions.

By definition, VR is a computer simulation of a real or imaginary system that allows a user to perform operations on the virtual (or simulated) system with the effects being rendered in real time. The CAVE is a room-size, high resolution environment with translucent walls, onto which stereo images are projected. Shutter glasses are worn which are synchronised with the projectors to rapidly alternate between the right and left eye, enabling the wearer to perceive depth. The CAVE also has trackers which monitor the user's head position and orientation, and updates the virtual image to the current view. Within a CAVE, navigation can be achieved by using a hand-held wand. The latter is a small hardware device with a pressure-sensitive joystick and trigger buttons. The wand

allows navigation within the environment and interaction with the virtual objects themselves. Gloves can be used for interaction but can also be extended to simulate a sense of touch. Since the CAVE provides full immersion and a large field of view, it is appropriate for the visualization and manipulation of large complex biological molecules.

During drug design, *in-silico* docking is being used for screening various protein and ligand interactions for the most stable and least energy interaction. To further help reduce the time taken for the drug discovery process, it is advantageous for experts to hand dock molecules before feeding the new molecular orientations to docking programs. An immersive environment provides better insight for docking and also exploits the user's intuition. Using experience and intuition, expert scientists can easily identify cavities in a receptor and manually place a ligand into the cavity. If this hand docked orientation is fed to a docking program, the search algorithm will ultimately take less time to find the optimal orientation, hence reducing computational time.

Some immersive programs have been developed for molecular docking, such as Vibe [2] and Stalk [3]. Vibe uses a High Performance Parallel Interface (HIPPI) [4] for a high-speed transfer, providing a real-time display of molecules in a virtual environment. Stalk, on the other hand, uses parallel, distributed, heterogeneous supercomputers with high-speed networks. However, one common problem is that these programs tend to focus more processing and data transmission.

2.2 Limited Collaboration

Another criterion worth considering while dealing with molecular visualization and docking, is the element of collaboration. Within the scientific community, there is often sharing of information and experience through collaboration in order to maximise the use of resources or expediate the generation of new ideas. Many scientists are not co-located and therefore collaboration becomes difficult. For instance, if two remote collaborators are working on the same molecular visualization, it would be difficult for each other to manipulate the molecule simultaneously. This is where collaborative environments would be useful. A collaborative session would enable multiple participants to simultaneously visualize or manipulate the same molecule. Similarly, collaborative docking of molecules could greatly enhance collaborative research. Few real-time collaborative computational programs exist. This is mainly because collaboration proves pointless when users are limited on visual information. More research is done on improving those programs' docking implementation rather than adding collaboration.

AMMP-Vis [5], [6] and AMMP-EXTN [7], [8] are among the few programs that have been developed to provide for an immersive visualization and protein modeling along with a collaborative session. AMMP-Vis is a collaborative multiview virtual environment for molecular visualization and modelling. The latter takes AMMP-Vis to a higher level by providing management of user privacy and cooperation. Unlike AMMP-Vis, AMMP-EXTN implements multiple parallel shared view sessions with different access policies. Thus, a user can create master

session which other participants can join. The problem with AMMP-Vis and AMMP-Extn is that both focus mostly on collaboration to the detriment of the docking algorithm being used.

2.3 Lack of Parallelization

Docking algorithms are implemented in two phases, namely: search and evaluation. The first step scans for the best configurational and conformational space, while the second implements a scoring function which rates and evaluates the different conformations. The search algorithms often include Monte Carlo (MC) [9], Genetic Algorithm (GA) [10] and Tabu search [11], among others, while the scoring functions often include force fields for the energies of the receptor-ligand interactions or the ligand's internal energy. For effective docking results, the search algorithm must cover 3D conformational spaces very quickly while the scoring function must quickly evaluate the different conformations among a huge population of poses (individual docked results).

In-silico docking algorithms consider docking to be either rigid or semi-flexible or flexible. In rigid docking, the bonds in both the ligand and the receptor have no freedom of movement. Semi-flexible docking algorithms, on the other hand, apply some flexibilities in the ligand's bonds, while flexible algorithms mimic more the way docking occurs in nature by allowing degrees of freedom for bonds in both the ligand and the receptor. One major problem with *in-silico* docking programs is the limitation of the docking algorithm. This is due to the fact that the use of complex algorithms such as MC and GA and the incorporation of more flexibilities in the algorithms involve a high level of computation and the use of massive computational resources to run the algorithms. To be able to deal with the computationally intensive docking algorithms and generate the results in feasible time, it is important to parallelize the whole process.

AutoDock [12] is a common program among the scientific community. According to a study carried out by Sousa et al. [13], in 2005 AutoDock was used by 27% of the scientists and since 2001 to 2005 AutoDock was ranked first among the five most commonly used docking programs, with an increased use from 36% in 2001 to 48% in 2005. Due to these various reasons and the fact that AutoDock is freely available, this docking program has been chosen for the initial phase of this project.

AutoDock can be used either for docking or virtual screening of substances. It comprises two different modules: AutoGrid and AutoDock. AutoGrid precalculates a set of grids describing the target protein, resulting in faster docking. These can be differentiated as grid maps and electrostatic map. While the AutoDock module performs the docking of the ligand to a set of grids. Configurational exploration is then performed. AutoDock uses both a MC algorithm and a modified version of GA, known as the Lamarckian Genetic Algorithm (LGA). Prior to running a docking job it is important to prepare a ligand and a receptor by adding hydrogens and associating atomic charges to atoms. This data is stored in a PDBQT format which is similar to the PDB [14], [15] format with an extra column to store partial atomic charges. Once AutoDock completes

a docking process, it generates a .dlg file which has the configuration of different poses ranked in order of energy interactions.

AutoDock is by default a sequential program, that is, the docking runs are executed one after the other. According to Khodade et al.[16], AutoDock takes 30 minutes to execute a docking run of 100-150 cycles. Thus, parallelization will significantly reduce the time taken for a docking task to complete. Different programs have been developed to perform parallelization of AutoDock, including the work of Khodade et al. and the program DOVIS [17].

Khodade et al. has modified the LGA algorithm in AutoDock to run in parallel. A population size of 50 with 100 and 200 runs resulted in a decrease from 81 minutes on an IBM Power-5 processor to only 1 minute on an IBM cluster of 96 processors. DOVIS, on the other hand, has a large-scale and high-throughput virtual screening technology. It uses the docking algorithm in AutoDock version 3 and runs parallel docking jobs through a queueing system on a Linux cluster and processes 500-1000 small compounds per processor on a daily basis.

3 System Description

In relation to the above problems identified, the proposed system uses immersive visualization and hand docking functionalities, along with robust docking algorithms from existing non-immersive programs, within a collaborative environment. Figure 1 depicts the system components and the interaction steps between them. The system comprises of different interfaces, namely: File Interface (FI), Visualization and Manipulation Interface (VMI), Docking Interface (DI) and Collaboration Interface(CI). The system components and the interaction steps between them are depicted in Figure 1.

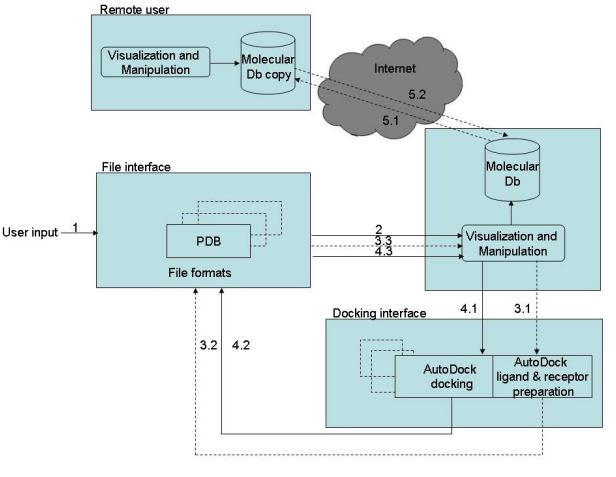


Fig. 1. System architecture

1. The system starts by accepting files from the user. Accepted file formats are PDB, PDBQT and .dlg. The PDB file has spatial data of molecules, while PDBQT and .dlg are file formats generated at an intermediate and final stage of AutoDock. The user also has the option to load multiple files in order to compare the complementarity of different ligands into the receptor's binding site. The FI reads the files which the user wants to load and also provides functionalities for saving the files in different formats.
2. VMI reads data from the file interface and projects the molecules in the CAVE. This module is being developed on VieGen [18]. Once files are loaded, the user can swap between different molecular representations, namely: Wire-frame, Stick, Ball-and-Stick and Space Filling/Corey-Pauling-Koltun (CPK). Different libraries have been created for this purpose. Manipulations can be done on the molecule, such as bond breaking and bond making. By using a wand, the user can easily rotate and move the receptor to find the cavity. Multiple ligands can be superimposed at the binding site for a clearer selection of the most appropriate ligand. Libraries of ligand fragments can be loaded and selected. These fragments can be manipulated and used for bond making purposes at the active site. Present work focuses on VMI and figure 2 shows the visualization of a CPK structural representations of the Histocompatibility Antigen (PDB id: 1HHI) in the CAVE. By rotating the structure, cavities can be easily spotted and ligands can be superimposed into a cavity.

   The correlation between VMI and DI is simply to allow a user to hand dock molecules by manipulating the molecules and then feed the data to the docking algorithm.
3. Prior to the docking interface, there is an optional module which is the preparation of the ligand and the receptor. This step depends on the type of docking program the user wants to run and is therefore optional. For instance, if the user wants to use AutoDock, the user can send the hand docked molecular conformation from the VMI to the DI (step 3.1 - please refer to Figure 1) and a PDBQT file will be generated, which can then be read by the FI (3.2) and visualized by the VMI (3.3).
4. DI reads data from VMI and runs the docking algorithm (4.1). As shown in Figure 1, initial emphasis will be laid on incorporating parallel AutoDock into the CAVE environment and visualizing the results in the CAVE (4.2 and 4.3). As both parallelized versions of AutoDock, described in the previous section, are freely available, it is planned to start with the program developed by Khodade et al. Later on this interface should enable plug-ins for configuring the different available docking programs to be used. The challenge involved is to be able to cater for all the different input file formats required by these docking programs and then incorporate the different stages that these docking programs might have prior to the actual docking procedure, such as steps 3.1 - 3.3 for AutoDock.
5. CI will be implemented using VieGen's networking functionalities. This interface has three points of contact with VMI, allowing remote users to visualize molecules loaded by the main user. The remote user can also request from

the server a copy of molecules prepared for docking (such as step 3.3 for AutoDock), hand docked configurations and the docking results generated by the chosen docking program(5.1). The remote user application makes a copy of the molecular data and also implements the FI. For manipulation purposes, a second copy of the molecular data is done. If the remote user modifies the spatial configuration of a molecule, the new molecular configuration data are compared with the first copy stored on the local database and only changed configuration data are sent back to the server. At the server's side, a copy of the modified molecular data is saved, without overwriting the original data (5.2). At any time the main user can view the different configuration changes done by the remote user. This is done by loading unchanged configurational data from the main user's molecular data and also by loading the modified configurational data from the data received from the remote user.

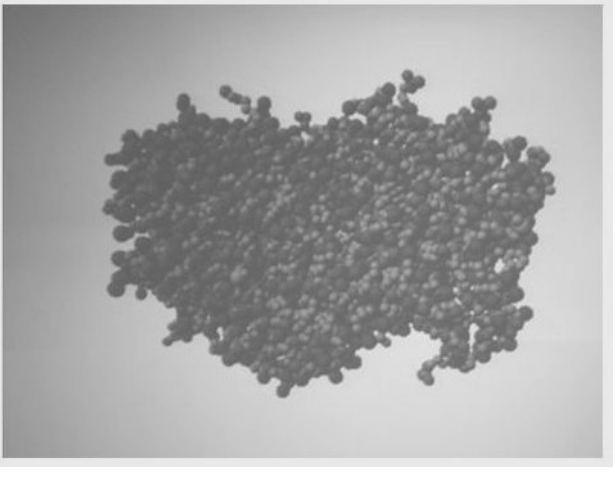


Fig. 2. CPK molecular visualization in CAVE

4 Conclusion

An immersive collaborative environment provides a high quality visualization which is beneficial for a thorough understanding and analysis of the 3D structure of molecules, particularly with respect to finding cavities and active sites in receptor molecules and trying to optimally hand dock ligands into those cavities. The CAVE provides an ideal environment to achieve such quality visualization. Once visualization and collaborative hand docking is achieved in the CAVE, robust algorithms in existing docking programs can be utilized to dock the new configuration from the user(s). Parallelizing docking programs cuts short docking algorithms running time. In this paper, a system has been described, which can

use hand docked configurations of a receptor and a ligand in a CAVE immersive environment as inputs to existing docking programs having robust and efficient docking algorithms. Molecular visualization in a CAVE has been achieved so far and integration of AutoDock is expected to be follow, along with a collaboration session of the system. Such a system would be useful in assisting drug design.

References

1. Cruz-Neira, C., Sandin, D.J., Defanti, T.A., Kenyon, R.V., Hart, J.C.: The CAVE: Audio visual experience automatic virtual environment Subsequences. Communications of ACM 35, 64–72 (1992)
2. Cruz-Neira, C., Langley, R., Bash, P.A.: VIBE: a virtual biomolecular environment for interactive molecular modeling Subsequences. Computers and Chemistry 20, 469–477 (1996)
3. Levine, D., Facello, M., Hallstrom, P., Reeder, G., Walenz, B., Stevens, F.: Stalk: An interactive system for virtual molecular docking Subsequences. IEEE Computations Science and Engineering, 55–56 (1997)
4. Tolmie, D., Renwick, J.: HIPPI: simplicity yields success, Subsequences. Network IEEE 7, 28–32 (1993)
5. Chastin, J., Zhu, Y., Brooks, J., Owen, S., Harrison, R.: A collaborative multiview virtual environment for molecular visualization and modeling. In: 3rd International Conference on Coordinated and Multiple Views in Exploratory Visualization (CVM), pp. 77–84. IEEE, London (2005)
6. Chastin, J., Zhu, Y., Brooks, J., Owen, S., Harrison, R.: AMMP-Vis: A collaborative virtual environment for molecular modeling. In: ACM Symposium on Virtual Reality Software and Technology, Monterey, pp. 8–15 (2005)
7. Ma, W., Zhu, Y., Harrison, R., Owen, G.S.: AMMP-EXTN: Managing user privacy and cooperation demand in a collaborative molecule modeling virtual system. In: Virtual Reality Conference VR 2007, pp. 301–302. IEEE, Los Alamitos (2007)
8. Ma, W.: AMMP-EXTN: a user privacy and collaboration control framework for a multi-user collaboratory virtual reality system. Masters Thesis, Georgia State University (December 2007)
9. Abagyan, R.A., Totrov, M.M.: Biased probability Monte Carlo conformation searches and electrostatic calculations and peptides and proteins Subsequences. J. Mol. Biol. 235, 983–1002 (1994)
10. Holland, J.: Adaptation in Natural and Artificial Systems Infrastructure. University of Michigan Press, Ann Arbor (1975)
11. Bland, J.A., Dawson, G.P.: Tabu search and design optimization Subsequences. Computer Aided Design 23, 195–201 (1991)
12. Goodsell, D.S., Olson, A.J.: Automated docking of substrates to proteins by simulated annealing Subsequences. Proteins: Structure, Function and Genetics 8, 195–202 (1990)
13. Sousa, S.F., Fernandes, P.A., Ramas, M.J.: Protein-ligand docking: current status and future challenges Subsequences. Proteins: Structure, Function and Bioinformation 65, 15–26 (2006)
14. Bernstein, F.C., Koetzle, T.F., Williams, G.J.B., Meyer Jr, E.F., Brice, M.D., Rodgers, J.R., Kennard, O., Shimanouchi, T., Tasumi, M.: Protein Data Bank: A computer-based archival file for macromolecular structures Subsequences. J. Mol. Biol. 112, 534–542 (1997)

15. Berman, H.M., Westbrook, J., Feng, Z., Gillilands, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The Protein Data Bank Subsequences. Nucleic Acids Research 28, 235–242 (2000)
16. Khodade, P., Prabhu, R., Chandra, N., Raha, S., Govindarajan, R.: Parallel Implementation of Autodock Subsequences. J. Appl. Cryst. 40, 598–599 (2007)
17. Zhang, S., Kumar, K., Jiang, X., Wallqvist, A., Reifman, J.: DOVIS: an implementation for high-throughput virtual screening using AutoDock Subsequences. BMC Bioinformatics 9, 126 (2008)
18. Haffegee, A.: VieGen: an accessible toolset for the configuration and control of virtual environments Ph.D Thesis, University of Reading (2008)

The gMenu User Interface for Virtual Reality Systems and Environments

Andrew Dunk and Adrian Haffegee

Centre for Advanced Computing and Emerging Technologies
The University of Reading
a.dunk@reading.ac.uk
http://www.acet.reading.ac.uk

Abstract. Desktop computers are able to provide a user interface with many features that allow the user to perform tasks such as execute applications load files and edit data. The gMenu system proposed in this paper is a step closer to having these same facilities in virtual reality systems. The gMenu can currently be used to perform a selection of common tasks provided by a user interface, for example executing or closing virtual reality applications or scenes. It is fully customisable and can be used to create many different styles of menu by both programmers and users. It also has shown promising results bringing some of the system based commands into the virtual environment, as well as keeping the functionality and adaptions required by applications. The use cases presented demonstrate a collection of these abilities.

Keywords: 3D User Interfaces, Virtual Reality.

1 Introduction

The development of quality 3D user interfaces and virtual environments can be complicated and time consuming. Currently there are no recognised standards on which to base the design and implementation of 3D user interfaces. Questions regarding interaction techniques, such as selections, travelling and way finding may have been answered already [1]. The area of environment and system control however, has not been paid as much attention.

Every computer needs an operating system that allows the user to utilise the computer hardware in an efficient way. These operating systems are combined with user interfaces to enhance its usability and usefulness. This research intends to be a step towards a three dimensional operating system for virtual environments. It proposes combining a customisable menu system, currently being developed as part of this research, with a virtual reality frameworks such as VieGen [2] which is currently being used for testing the gMenu system. This combination will supply equivalents for the more common operations provided by today's desktop operating systems. These will include simple file browsing, application execution, system controls (such as audio settings), as well as being able to perform basic object manipulations within an environment (including scaling, rotations, and constraining movement to a specific axis).

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 746–753, 2009.

This paper starts by introducing a background on existing user interfaces, then continues by elaborating on the gMenu's current and proposed features. Several use cases are described, and finally some conclusions are drawn and future work is proposed.

2 Virtual Reality User Interfaces

In order to compile a standard for 3D user interfaces, studies need to be made on existing 3D user interfaces which have been designed for specific applications, and have gradually improved over time as new requirements and issues where found. There has been significant research into comparing and classifying these user interfaces to gain a overall understanding of the requirements of a useful and usable user interface. [1,3]

2.1 Interaction Techniques

There are many interaction techniques available for 3D user interfaces that allow users to navigate through environments, as well as select and manipulate objects.

Selection is an integral part of using a menu and there are many different methods available for this; for example, collision detection is one of the simplest methods of selection, the user would reach out and touch the object to select it. This becomes difficult if the objects are placed at further distances. To overcome this ray casting selection could be used; this method simulates a laser pointer extending the selection to any distance.

Other selection methods include occlusion which works by concealing the object to be selected. This method is relative to your eye and hand position. [4] There are also selection techniques that use eye tracking only, enabling selections to be made based on the users gaze rather than their hand position. [5]

2.2 System and Environment Control Menus

3D menu systems exist in many forms and designs. Types of menus include 2D menu systems that have been brought into 3D environments, 3D menus positioned at specific locations relative to the environment, the users, or objects, and menus that are specific to hardware devices used with the virtual reality system.

There are many 2D graphical user interfaces available, one solution is to use these systems and adapt them to the 3D environment. This gives advantages to usability as systems like Windows Icons Menus and Pointers (WIMP) have had many years of research and improvements applied to them. They are very well recognised and would require a minimal amount of training for users to confidently use them.

A variety of pop-up and pull-down style menus have been designed and studied in virtual environments [6], as well as implementations enabling the X window system to be brought into a virtual environment [7]. Menus have also been defined

by their position in the environment, the tool belt menu was displayed around the users waist while in the environment and hand held menus such as the ring menu is displayed by the users hand while being used.

A multitude of menus have been designed for use with particular hardware in mind like the TULIP menu [8] which use pinch gloves. Menus can be projected onto physical objects like tablets which in turn give a tactile feedback as well as visuals when being used. [9]

Other menu systems include the Spin Menu [10], designed for quick access and uses 3D icons and the Command and Control Cube [11] used on the holobench. These two menus use an action or movement to select menu items rather than the idea of pointing and selecting.

3 The gMenu System

This section is split into three parts. The first describes some of the proposed and current abilities of gMenu and how it can be constructed and configured. The second describes the menu as a file browser and how it is able to create icons for specific file types. The third section describes messaging between the menus, objects within a scene, and the virtual environment in conjunction with the VieGen framework.

3.1 A gMenu

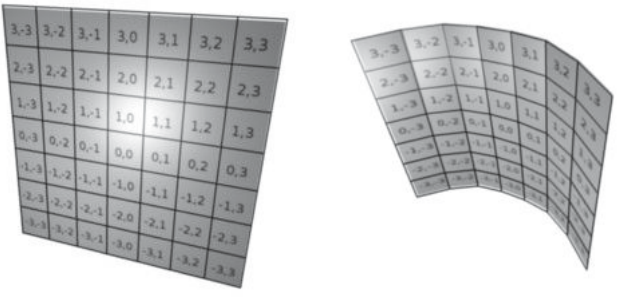
Each new menu is constructed with a menu grid, which is made up of multiple menu items. It is possible to create new menus within an existing menu producing submenu functionality. Submenus automatically link back to their parent when created.

Multiple gMenus can be created in a virtual environment, each gMenu could be organised to give better clustering of the commands used within the virtual environment.

The menus will be able to be created by programmers and customised by users. Simple XML files will be able to be written by users to generate custom menus and link there functionality to the system and environment. An XML parser will be used to generate the user created menus and bring them into the environment. The programmer will have a powerful library to configure all aspects of a created gMenu.

gMenu Items. gMenu items are created and inserted into locations on a gMenu grid. It is possible to develop new gMenu items by extending a base gMenu item container. Currently there are three types of items that are available, these are a textured button which is a cuboid object with textured surfaces, a 3D item which is a detailed 3D model object rather than an flat texture, and a 3D text string for displaying text within the menu.

Each item has some base properties including its position in the grid, its visibility, if it is enabled, highlighting function, states, and messaging options



(a) Flat Plane Grid (b) Curved Wall Grid

Fig. 1. Examples of Possible Grid Positioning with Associated Coordinates

for communicating the states and commands throughout the system. When a menu is being displayed within an environment, each of the items are frequently checked for collisions with the users interface device. Items are able to send messages describing their current state. for instance if the user is hovering over the item, pressing a button on the users interface device over an item, or releasing a button on the users interface device while over an item.

If a hover state is triggered then a highlighting function is applied to the item. The default highlight translates the item forward towards the user by the same depth as the item, giving the effect of the button being selected. However the highlight functions can be replaced allowing the creator to develop their own customised functionality.

The gMenu Grid. The gMenu grid is designed to give the items created within the menu a defined structure and keep the visual aspect of the menu clean. This allows the creator to concentrate on the functionality of the menus rather than their appearance.

The grid is never displayed and only exists as a function that generates a position in 3D space for a given (x, y) item coordinate. To place an item into the grid simply give the item an (x, y) coordinate, which is translated into the position of the item in 3D space. This also allows new grid generation functions to be written, meaning the menus physical shape and style can be customised. A flat plane Fig.1(a) or a curved wall Fig.1(b) are possible examples of grids that can be created for positioning each of the items. Items do not have to exist on a single plain but they are dependant on the grid. Items can be placed at any location within the grid allowing items to be clustered and arranged into sections and different shapes Fig.2(b).

The size of the cells in the grid and their separation can be set when creating the menu grid, this denotes the maximum size of the menu items and the distance they are placed apart from one and another. The grid itself can be as large as

required. However considerations need to be made if creating large grids, at the efficiency of searching for an item within a large grid could possibly produce poor results.

3.2 The gMenu File Browser

The gMenu file browser extends the functionality of the base gMenu system to create an iconic display of a file system starting from a root directory.

When a gMenu file browser is created a recursion of the file system is performed, each file and folder beneath the root path is read and an item is created within the file browser grid. Submenus are created for each directory with all appropriate files included as separate items. As each item is added to the menu a textured button is created and supplied a texture to represent the file or folder. In addition a text item is created to identify each icon by displaying the file or folder name.

Some file types are recognised by the file browser, these are given specific icons. If supported image files are found within the file system the image will be used as the buttons texture. Recognised 3D model files are either added to the menu as a scaled version of the 3D model, or a textured button can be displayed with an image created from the 3D model file. By creating images for the 3D models the system load is reduced. Any model images created are saved in the same directory as the 3D model to decrease load times on consecutive runs. There is also support for basic filtering of file extension types, making it possible to create file browsers of specific file types only. If a folder is selected the corresponding submenu will replace the current menu, if an item is selected, the items file name and location are returned.

Creating Button Textures. If a button texture is needed for a 3D model a set of steps are taken. Firstly a search is performed in the same directory as the file for a previously created texture associated with the file, if one is found that texture is simply returned as the files icon. If a texture file is not found then a new one is created.

This is performed by loading the model into a predefined scene that is never rendered on the users display. The scene has some basic settings that can be customised, such as the size of the viewport being used which determines the end size if the icon, and the scenes background colour. Once the 3D model has been loaded into the scene a camera is positioned to capture the entire model and an image is rendered and saved in the same directory as the model so that the texture will not need to be generated again, saving load time. Once the image has been saved it is returned to be used in the menu item.

3.3 VieGen Specific Implementation

Within VieGen, the different entities that make up a virtual environment are all derived from the base SceneObject class, which provides them with in-scene

attributes and behaviours. All SceneObjects have the ability to produce and consume messages, which are used to provide communication and interaction amongst these entities. Typical VieGen messages include those such as the user selecting an object, or notification that one object has collided with another. These messages provide the basis for interacting with the gMenu system. Focus changing or selection events can be used to indicate changes in the gMenu Item state, and the generation of new messages could result from these interactions. While some items could be hard coded to provide specific actions, others could be configurable. It the latter case, options could range from sending named system events, through to the flexibility of sending XML fragments for direct object control.

4 Applications

This section describes some simple applications using the gMenu and shows its current functionality.

4.1 3D Model Viewer

To demonstrate the gMenus file browser features, a simple environment has been created that allows users to brows through a file hierarchy and load selected 3D models into the environment, position the models within the environment and change their orientation. The environment itself is a small area with no visible walls or floor, it contains four lights to equally illuminate the area and has a blue background colour.

The gMenu file browser can be opened in the environment by pressing a button on the users wand. The menu appears a short distance in front of the wand position and a ray casting technique is used to select items within the menu. The file menu is filtered to only show supported 3D model files. Textures have been generated for each of the models and predefined textures are loaded for directories and a back button of each subdirectory. These icons are a folder and an arrow pointing to the left respectively.

By hovering over an item, text is displayed with the model's name; when the item is selected the model file is loaded and attached to the users wand so that it can be positioned in the environment. Any number of models and duplicates can be loaded into the environment.

4.2 The Virtual Keyboard

This application shows the visibility of the gMenu system by creating a full keyboard layout in front of the user, enabling text input while in a virtual environment Fig.2(a).

The user can type each letter by moving their hand to collide with the appropriate button. It would be possible to "two finger type"using this gMenu keyboard if two hand trackers were used within the environment.

(a) gMenu Keyboard (b) Custom Item Layout

Fig. 2. Examples of the gMenu system in use

The keyboard can be very useful for naming files, text messaging other users within networked environments or debugging applications still under development while still within the environment. This could be achieved by generating a virtual terminal.

4.3 Environment Controller

This application combines a file browser with other control options to enable users within a virtual environment to load, edit, and save scenes without having to leave the virtual reality system. Scenes are loaded in the same way as the 3D model viewer, but the scene objects can be selected bring up a control menu to perform simple manipulations of the objects. The scene can then be saved and closed.

Scenes that incorporate audio streams allow extra options in the control menu to change the volume within the environment.

5 Conclusion and Future Work

The gMenu system is very easy to create as a programmer and customise as a user. Because of these abilities it is very easy to simulate many types of 3D menus that have already been developed and researched, and will also enable quick prototype designs of new new menu ideas to be produced and tested in the future.

There has been positive reactions towards usability from the initial tests and applications created using the gMenu system. It shows potential to be used in multiple application tasks with its ability to be highly customisable and styled. Future user testing has been planned for the gMenu system comparing its usefulness and usability with other menu systems.

With a fully customisable system like gMenu and the ability to control both a system and its environments brings us a step closer to some of the capabilities that are possible with desktop operating systems.

There are plans to introduce a third dimension to the gMenu grid allowing developers more freedom with the menu shapes and positions, making the option to generate menus resembling the Command and Control Cube easier for example.

Customising and creating new menus live within the environment by adding and removing items from menu grid to grid, or even placing menu items into the environment on there own, are some ofthe features currently being developed.

There is also future plans to create more menu item types to be used in gMenu such as toggle buttons, passive indicators, analogue output selections, and dynamic items that can be created by the users.

References

1. Bowman, D.A., Kruijff, E., LaViola, J.J., Poupyrev, I.: 3D User Interfaces: Theory and Practice. Addison Wesley Longman Publishing Co., Inc., Redwood City (2004)
2. Haffegee, A.: VieGen: An Accessible Toolset for the Conguration and Control of Virtual Environments. Ph.D thesis, Centre for Advanced Computing and Emerging Technologies (ACET), School of Systems Engineering, University of Reading (2008)
3. Dachselt, R., Hbner, A.: Three-dimensional menus: A survey and taxonomy. Computers and Graphics 31(1), 53–65 (2007)
4. Bowman, D.A.: Interaction Techniques for Common Tasks in Immersive Virtual Environments. Ph.D thesis, Georgia Institute of Technology (1999)
5. Chan, C.N., Oe, S., Lin, C.S.: Active eye-tracking system by using quad ptz cameras. In: Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE, pp. 2389–2394 (November 2007)
6. Jacoby, R.H., Ellis, S.R.: Using virtual menus in a virtual environment. In: Alexander, J.R. (ed.) Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 1668, pp. 39–48 (June 1992)
7. Coninx, K., Van Reeth, F., Flerackers, E.: A hybrid 2d/3d user interface for immersive object modeling. In: Proceedings of Computer Graphics International, 1997, pp. 47–55 (June 1997)
8. Bowman, D.A., Wingrave, C.A.: Design and evaluation of menu systems for immersive virtual environments. In: VR 2001: Proceedings of the Virtual Reality 2001 Conference (VR 2001), Washington, DC, USA, p. 149. IEEE Computer Society, Los Alamitos (2001)
9. Wloka, M.M., Greenfield, E.: The virtual tricorder. Technical report, Department of Computer Science, Brown University (1995)
10. Gerber, D., Bechmann, D.: The spin menu: a menu system for virtual environments. In: Virtual Reality, 2005. Proceedings. VR 2005, pp. 271–272. IEEE, Los Alamitos (2005)
11. Grosjean, J., Burkhardt, J.M., Coquillart, S., Richard, P.: Evaluation of the command and control cube, p. 473. IEEE Computer Society, Los Alamitos (2002)

VIII International Workshop on Computer Graphics and Geometric Modeling - CGGM'2009

Andrés Iglesias

Department of Applied Mathematics and Computational Sciences,
University of Cantabria, Avda. de los Castros, s/n, E-39005, Santander, Spain
iglesias@unican.es
http://personales.unican.es/iglesias/

Abstract. This short paper is intended to give our readers a brief insight about the Eight International Workshop on Computer Graphics and Geometric Modeling-CGGM'2009, held in Baton Rouge, Louisiana (USA), May 25-27 2009 as a part of the ICCS'2009 general conference.

1 CGGM Workshops

1.1 Aims and Scope

Computer Graphics (CG) and *Geometric Modeling* have become two of the most important and challenging areas of Computer Science. The CGGM workshops seek for high-quality papers describing original research results in those fields. Topics of the workshop include (but not limited to): geometric modeling, solid modeling, CAD/CAM, physically-based modeling, surface reconstruction, geometric processing and CAGD, volume visualization, virtual avatars, computer animation, CG in Art, Education, Engineering, Entertainment and Medicine, rendering techniques, multimedia, non photo-realistic rendering, virtual and augmented reality, virtual environments, illumination models, texture models, CG and Internet (VRML, Java, X3D, etc.), artificial intelligence for CG, CG software and hardware, CG applications, CG education and new directions in CG.

1.2 CGGM Workshops History

The history of the CGGM workshops dates back nine years ago, when some researchers decided to organize a series of international conferences on all aspects of computational science. The first edition of this annual conference was held in San Francisco in 2001 under the name of *International Conference on Computational Science, ICCS*. This year ICCS conference is held in Baton Rouge, Louisiana (USA).

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 757–758, 2009.

After ICCS'2001, I realized that no special event devoted to either computer graphics or geometric modeling had been organized at that conference. Aiming to fill this gap, I proposed a special session on these topics to ICCS'2002 organizers. Their enthusiastic reply encouraged me to organize the first edition of this workshop, CGGM'2002. A total of 81 papers from 21 countries were submitted to the workshop, with 35 high-quality papers finally accepted and published by Springer-Verlag, in its Lectures Notes in Computer Science series, vol. 2330. This great success and the positive feedback of authors and participants motivated that CGGM became an annual event on its own. Subsequent editions were held as follows (see [1] for details): CGGM'2003 in Montreal (Canada), CGGM'2004 in Krakow (Poland), CGGM'2005 in Atlanta (USA), CGGM'2006 in Reading (UK), CGGM'2007 in Beijing (China) and CGGM'2008 again in Krakow (Poland). All of them were published by Springer-Verlag, in its Lecture Notes in Computer Science series, volumes 2668, 3039, 3515, 3992, 4488 and 5102 with a total of 52, 24, 22, 22, 20 and 16 contributions, respectively. In addition, one Special issue has been published in 2004 in the *Future Generation Computer Systems - FGCS* journal [2]. Another special issue on CGGM'2007 is available online and in the way to be published in printed form soon.

2 CGGM'2009

This year CGGM has received a total of 11 papers of which 6 have been accepted as full papers. The reader is referred to [3] for more information about the workshop. The workshop chair would like to thank the authors, including those whose papers were not accepted, for their contributions. I also thank the referees (see the CGGM'2009 International Program Committee and CGGM'2009 International Reviewer Board in [3]) for their hard work in reviewing the papers and making constructive comments and suggestions, which have substantially contributed to improving the workshop.

Acknowledgments

This workshop has been supported by the Spanish Ministry of Education and Science, National Program of Computer Science, Project Ref. TIN2006-13615 and the University of Cantabria. The CGGM'2009 chair also thanks Dick van Albada, workshops chair of general conference ICCS'2009 for his all-time availability and diligent work during all stages of the workshop organization. *Thanks for your support, Dick!*

References

1. Previous CGGM: `http://personales.unican.es/iglesias/CGGM2009/Previous.htm`
2. Iglesias, A. (Guest ed.): Special issue on "Computer Graphics and Geometric Modeling. Future Generation Computer Systems 20(8), 1235–1387 (2004)
3. CGGM 2009, `http://personales.unican.es/iglesias/CGGM2009/`

Reconstruction of Branching Surface and Its Smoothness by Reversible Catmull-Clark Subdivision

Kailash Jha

Assistant Professor, Deptt. of Mechanical Engineering
& Mining Machinery Engineering,
Indian School of Mines University, Dhanbad-826004, Jharkhand, India
kailash_jha@hotmail.com

Abstract. In the current research a new algorithm has been developed to get surface from the contours having branches and a final smooth surface is obtained by reversible Catmull-Clark Subdivision. In branching, a particular layer has more than one contour, corresponds with the contour at the adjacent layer. The layer having more than one contour is converted into a 3D composite curve by inserting points between the layers. The points are inserted in such a way that the center of contours should merged to the center of the contours at the adjacent layer. This process is repeated for all layers having branching problems. In the next step, 3D composite curves are converted into different polyhedrons by the help of the contours at adjacent layers. Number of control points at different layer for contours and 3D curves may not be the same, in this case a special polyhedron construction technique has been developed. The polyhedrons are subdivided using reversible Catmull-Clark subdivision to give a smooth surface.

Keywords: Catmull-Clark subdivision, branching surface, incompatible curves, reconstruction.

1 Introduction

In the present work, an algorithm has been developed to construct a three-dimensional surface from contours at different layers which may have branching problem and the required smooth surface is obtained with the help of the reversible Catmull-Clark subdivision. In the reversible Catmull-Clark subdivision level of smoothness is given by an integer value for required rendering. Construction of 3D surface from 2D contours is very important for CAD (Rapid prototyping, NC machining), Medical imaging and Geographical Information System. Technologies such as magnetic resonance imaging (MRI), computed topography (CT), and ultrasound imaging allow measurements of internal properties of objects to be obtained in a nondestructive fashion.

The points on the 2D contours are measured slice-by-slice. A slice may have more than one independent closed contour. These contours may correspond to one or more contours at adjacent slice. This situation is termed as branching, which is described in this work. The set of planes generating the slices are usually parallel to each other and may not be equi-spaced along any axis through the object. Once these slices have

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 759–769, 2009.

been obtained, the goal is to enable a human to easily visualize in 3D. Many algorithms have been developed for this purpose, but they can be classified into two categories, Volume Rendering Methods and Surface Reconstruction Methods. Volume rendering is used to show the characteristics of interior of the solid. In surface rendering, a geometrical representation is used to model the object or structure to be visualized based on original data such as edge, mesh, polygon, triangle or pixel. The present work focuses on surface reconstruction method. Generation of 3D surface from 2D contours has four basic steps: (a) Correspondence, (b) Tiling, (c) Branching and (d) Generation of surface.

(a) The correspondence problem involves finding the correct connections between the contours of adjacent slices. In the present formulation the correspondence problem is known which can be obtained from [1].
(b) Tiling means using slice chords to triangulate the strip lying between contours of two adjacent slices into tiling triangles. A slice chord connects a vertex of a given contour to a vertex of the contour in an adjacent slice. Each tiling triangle consists of exactly two slice chords and one contour segment. Details of tiling can be obtained from [1, 2]. In the current work focus is given on branching problems. Our approach does not require tiling because it is based on construction of polyhedrons.
(c) A branching problem occurs when a contour in a layer corresponds to more than one contour in an adjacent layer.
(d) Solution of branching problem results polygonal surface which are smoothened by reversible Catmull-Clark subdivision.

There are three types of branching problems: (1) One-to-one, (2) One-to-many and (3) Many-to-many. One-to-one problem has been solved by several researchers based on minimizations of energy, twist and curvature and different tiling techniques which are shown in Fig 1a. One-to-many problem is the one in which at least one layer must have more than one contour and have correspondence with the contour at adjacent layers which is shown in Fig 1b. Many-to-many problem is stated as *m* contours at *i-th* layer and *n* contours at *(i+1) th* layer and they are corresponding to each other. This problem can be solved by combining many one-to-many branching problems. Fig. 1c shows the Many-to-many branching problem. In the present technique, control points of curves at layer having branching problem are taken at a time and converted into 3D curve by linear merging of geometrical centers of the contours toward the center of contours at adjacent layer. For example, two contours C_1 and C_2 are shown

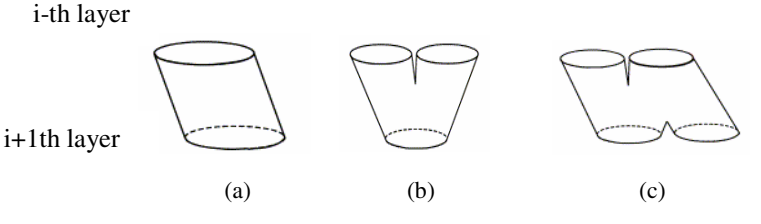


Fig. 1. Different type of branching: (a) One-to-one b; (b) Many-to-one; (c) Many-to-Many

in Fig. 2 at layer *i* having geometrical centers at g_1 and g_2 respectively, which are merged linearly to the geometrical center (g_3) of adjacent contour C_3 at *(i+1)th* layer. Lines parallel to g_1g_3 are drawn from all the control points of curves C_1 and similarly lines parallel to g_2g_3 are drawn from control points of curve C_2. The intersection point having highest value of y-coordinate is considered for construction of 3D curves which is point d in Fig 2. ad and bd are parallel to g_1g_3 and g_2g_3 respectively and intersect at highest value of y-coordinate to give 3-D curve, which is comprised of db, C_2, bd, da, C_1 and ad respectively. Once 3-D curve is obtained and the remaining contours at other layers do not have any branches, then the polyhedrons are constructed according to present formulation given in section 4 and the final 3-D surface is obtained by reversible Catmull-Clark subdivision, otherwise same process is repeated. Fig. 2 shows the contour points as well as other additional points. The current branching problem has been solved for the known correspondence and starting points for the contours at different layers. A complex 3D contour is generated for layer having branching problem, which converts many-to many branching problems into one-to-one branching problems.

Polyhedrons are constructed for all the pairs of adjacent layers once there is no branching problem. If contours are given as the points and constraints then the control polygons can be obtained by the references [3, 4] using energy based curve approximation.

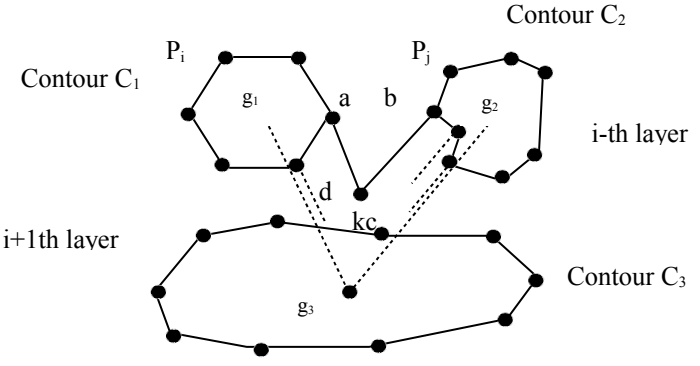


Fig. 2. Illustrated example to convert contours to a 3-D composite contour for a layer having two contours

A polygonal complex is a simple polygonal mesh whose structure depends on subdivision scheme used and whose limit of subdivision is a curve rather than a surface. The polygonal complexes are constructed for each of the polygons and polyhedron is constructed by connecting the polygon complexes side-by-side. Contours given are restricted to uniform cubic B-Spline curves [5, 6] but it can be overcome by non-uniform technique given in [7]. The proposed technique is of order O (kn), where n is the number of cross section curves and k is the average number of control points per curve in the given sectional curves.

The reversible Catmull-Clark subdivision has been developed in the present work to get smooth surface from a given polygons. It can be used for mesh generation. In graphics, shading and rendering improves the representation of surfaces and solids. The polygon construction technique in the current work is similar to [5, 6], which is not limited to the uniform B-Spline input curves. In the present work other type surface subdivision [8] can be incorporated. Reversible Catmull-Clark subdivision has been achieved in the present work, which is a unique feature of this work and is different from [6]. It also gives different stage of convergence of polygons toward the interpolating curves as well as smoothness of the surface. A boundary curve of a B-Spline patch is dependent only on the first three rows of the mesh defining the patch. It is same for the curve case where one end depends on the three control vertices. Present technique will be useful in medical imaging like modeling liver vessel tree, geological structure and stem of the tree.

Previous works are given in section 2. A brief overview of Catmull-Clark subdivision is given in section 3. Section 4 describes the current work. Results and discussions have been given in section 5. The research has been concluded in section 6.

2 Previous Works

The literature that is devoted to many-to-one branching problem can be classified into four main families. The family of contour connection methods attempted to artificially render one-to-many problem into a one-to-one by connecting the disjoint contour with line [2] or triangulate facet bridge [9]. The first choice is applicable for simple cases, while the second constrains unnaturally the saddle points of the branching surface to lie on the plane containing the disjoint contours. The second family is based on introduction of intermediate contour, which splits the original problem into two problems one-to-one and a new one-to-many. The second problem is further simplified into m one-to-one problems. This idea has been proposed in [10, 11] and has been implemented in [12]. The family of partial contour connection and hole filling has been proposed in references [1,13,14]) is characterized by matching partially the disjoint contours with the single contours of the neighboring plane, thus leaving a number of holes which are finally filled in final step. Goodman et al., [15] has treated one-to-two case in which a single hole is filled by an approximated chosen hyperboloid.

Finally, the family of implicit schemes relies on the assumption that possesses implicit representation of contours composing the cross-sections. Then an implicit interpolate can be obtained by taking a convex combination of contour representation [16], or implying the distance function [12, 16]). Wang et al., [17] used Catmull-Clark subdivision for biorthogonal wavelet construction based on lifting scheme. Loop and Schafefer, [18] approximated Catmull-Clark subdivision surfaces by minimal set of bicubic patch. A brief overview of branching problems has been explained in reference [19], which involves skinning, trimming and hole filling. In reference [20] an algorithm has been developed to obtain branching surfaces by energy based skinning of compatible 2-D curves obtained by energy-based approximation.

3 Catmull-Clark Subdivision

A polygonal mesh from which inner mesh can be obtained through the application of following rules basically defines a Catmull-Clark surface:

1. Each old face f with n vertices $(V_i)_{i<i<n}$, a new vertex V_f can be generated at the centroid by:

$$V_f = 1/n \sum_{i=1}^{n} V_i \tag{1}$$

2. For each of old edge e having two vertices V_1 and V_2 which is shared by two faces f and g, a new vertex ve can be generated by:

$$V_e = (V_1+V_2+V_f+V_g)/4 \tag{2}$$

3. For each old vertex V incident to n edges (e_i) and shared by n faces f_i, a new vertex V_v can be generated by:

$$V_v = \alpha_n \sum_{i=1}^{n} V_{ei} + \beta_n \sum_{i=1}^{n} V_{fi} + \gamma_n V \tag{3}$$

Where V_{ei} (respectively V_{fi}) is the vertex generated from the edge e_i (respectively the face fi), and the weight α_n, β_n and γ_n are given by:

$$\alpha_n = \beta_n = 1/n^2, \quad \gamma_n = (n-2)/n$$

The limiting curve of a Catmull-Clark polygonal complex can be determined in a piecewise manner. Basic formulation for Catmull-Clark subdivision has been given in [21]. Surface subdivision is based on rules and the given input polygons. Fig 3 shows the polygon and the subdivided polygons have been shown in Fig 4 along with the modified vertices, edge points and face point.

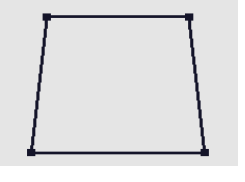


Fig. 3. Constructed polyhedron

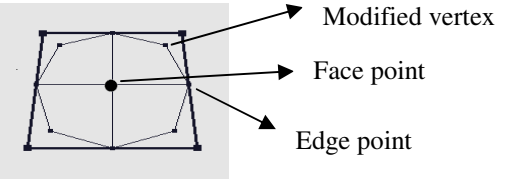


Fig. 4. Subdivided polyhedron with face points and vertices

4 Current Work

The present technique is based on construction of polyhedron from the generated 3D composite curves and polygons. Two or more set of control polygons are given at a layer which correspond to the control polygons of curves at adjacent layer. If the control points of the contours are not given, energy based approximation [3, 4] can be

used to find their value. The first task is to calculate the center of each closed polygons and find slope of the line joining the centers of corresponding contours. In Fig 2, g_1, g_2 and g_3 are the CGs of the control polygons and $\underline{g_1g_3}$ and $\underline{g_2g_3}$ are the line joining their centers. An algorithm has been developed to find out points (P_i, P_j) in given curves for which the intersection of line parallel to g_1g_3 and passes through point p_i and line parallel to g_2g_3 and passes through q_j having highest value of y-coordinate intersection. Points P_i, P_j and the intersection point d have been shown in Fig 2. Once the intersection point is obtained which is between the contours having correspondence, the closest point from this point at the adjacent layer is determined which, is the point *kc* in Fig 2. With reference to these two points (d, kc), polyhedrons are constructed according to algorithm given in section 4.1. If the number of control points in the 3D composite curves is not the same as the number of control points at adjacent layer of contour then they are considered as the incompatible curves, which are solved by the technique given in ref [5, 6]. After having the correct polyhedrons, the smooth surfaces are obtained by reversible subdivision of the polyhedrons (Catmull-Clark Subdivision), which is explained in Section 2. In the present work, reversible subdivision techniques have been adopted to go for the required smooth surface. Level of smoothness can be given by an integer value. Once the level of smoothness is given by the user, the corresponding surface will be displayed on the screen.

4.1 Construction of Polyhedrons for Branching Contours

The construction of surface polyhedron is done in such a way that a final control mesh will be obtained by connecting the resulting complexes side-by-side. This construction is divided into two phases:

(a) Reversible subdivision phase, (b) decomposition phase

These terminologies have been explained in references [5, 6]

4.1.1 Reversible Subdivision Phase

In this phase the given control polygons (M_i and M_j) (see Fig 5) of input curves are subdivided recursively until one of the subdivided control polygon does not have one edge. There are n and m vertices in control polygons M_i and M_j respectively. The ends of the polygons are joined together to make closed polygon. This closed polygon is again divided into two polygons by connecting at points where its Euclidian distance is minimum. Fig 5 shows the connection. This phase is also called virtual primitive phase. Fig 6 shows a primitive face, several such faces will be the output of the reversible subdivision phase.

Recursive subdivision can be illustrated by the following points:

1. V_o, V_n V_m and W_m are the boundary vertices
2. If m and n are one stop
3. Associated vertices V_p and W_q are calculated by shortest distance criterion
4. Subdivision is continued recursively for divided polygons

This process terminates with a set of virtual primitives faces, where each such face f is represented by the pairs (vf, wf) with vf and wf are the two set of vertices $V_f = (V_i) a \le i \le a+r$ and $W_f = (W_i)\ a \le i \le a+s$ characterized by the following properties:

1. Either r and s is equal to 1
2. The vertices v_a and w_b are the closest to v_{a+r} and w_{b+s} respectively among all possible pairs (V_i, W_i) of the set $(V_f X W_f)$

A primitive face F can be defined by sequence T from the vertices M_i and another sequence B from the vertices M_j. One of the sequences must have two vertices. Let B be the sequence having two vertices

$$B = \{W_1, W_2\} \quad (4)$$

$$T = \{V_1, V_2 \quad V_k\}$$

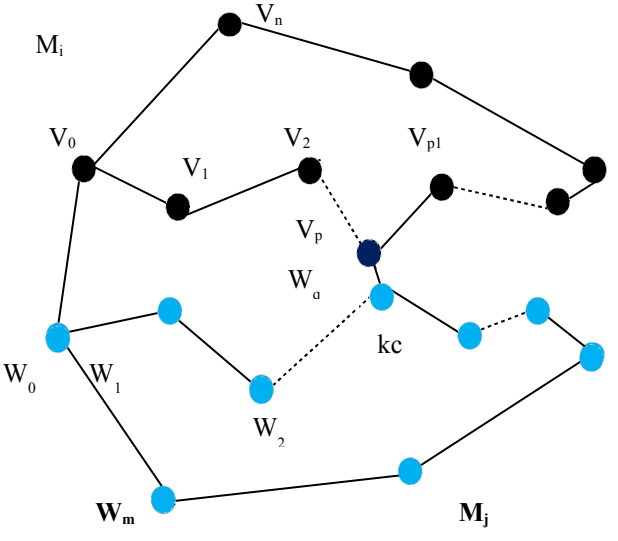


Fig. 5. Illustration for recursive subdivision

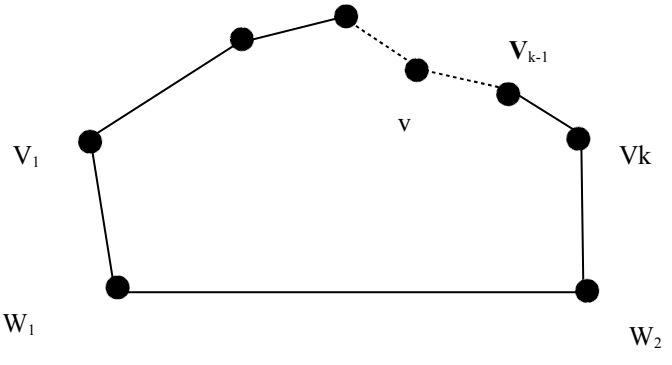


Fig. 6. Primitive face obtained by recursive subdivision for decomposition

4.1.2 Decomposition Phase

In this phase the primitive faces are decomposed into several actual faces. Number of the faces is equal to number of vertices (k) in T. First the k-2 vertices are created on B between W_1 and W_2 in same proportion distance as the vertices of V_k. K vertices (V_k') are created in between the two set of vertices B and T in 2: 1 ratio of distance and similar two vertices (W_1' and W_2') are created on line $\underline{V_1W_1}$ and $\underline{V_nW_n}$ in 1:2 ratio of distance. Constructed polyhedrons and surfaces are shown in Fig 7. $V_1V_2V_2'V_1'$ is a typical constructed polyhedron in Fig 7. Constructed polyhedrons for non-branching incompatible curves and has been shown in Fig 8.

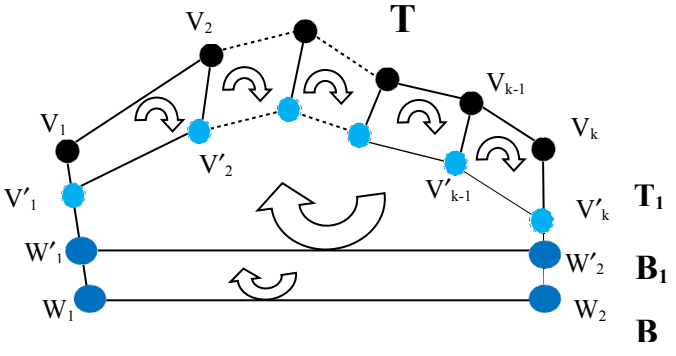


Fig. 7. Final faces after decomposition of a primitive face

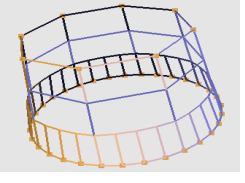


Fig. 8. Polyhedrons for incompatible contours

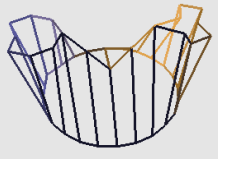


Fig. 9. Polyhedrons for branching contours

5 Results and Discussions

A large number of results have been tested in this work for contours having branching problems. The data structure implemented supports recursive Catmull-Clark subdivision. C++ and OpenGL have been used in current formulation. Figure 9 shows the constructed polygons for a branching problem in which a layer has two contours and they correspond to a contour an adjacent layer. Fig. 10 shows a wire frame surface obtained by the present algorithms after smoothening by reversible Catmull-Clark subdivision. In this test result, there are three contours at a layer and they are corresponding to a contour at an adjacent layer.

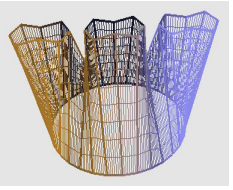


Fig. 10. Wire frame surface with constructed polyhedrons

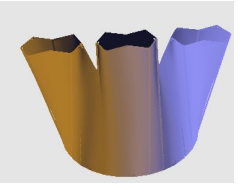


Fig. 11. Shaded branching surface

Fig 11 shows the shaded surface for the same test result. Another test result is shown in Fig 12 which has three contours at a layer and two contours an adjacent layer. Fig 13 shows the shaded surface for the test result shown in Fig 12. Fig 14 shows another test result-having contours at seven layers, the first and the last layer having two contours and it is correspondence with contours at adjacent layers. Fig. 15 shows the shaded surface for the test result shown in Fig 14. The technique given in [5, 6] can not directly handle the branching surface. The results given in [19] for branching are shifted towards the junction of the layer having single contour.

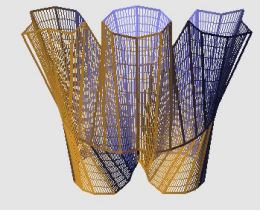


Fig. 12. Wire frame branching surfaces

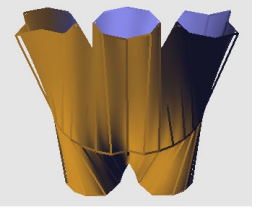


Fig. 13. Shaded surface & constructed polyhedrons

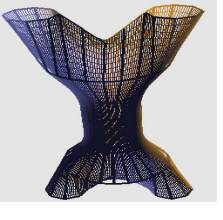


Fig.14. Subdivided wire frame branching surface

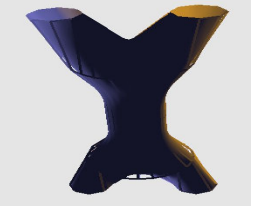


Fig. 15. Shaded branching surface with constructed polyhedrons

6 Conclusions

An algorithm has been developed to get a smooth surface from contours having branching problem. Three-dimensional composite curves have been obtained for the contours having branching problem on the basis of merging of the centre of the contours towards the center of adjacent contours. The polyhedrons have been constructed

between the 3D composite curves and other contours. Two adjacent contours which are free from branching problem are selected for construction of polyhedrons. Reversible Catmull-Clark has been designed in the present formulation for the required smooth surface for better rendering. Different results for branching problem have been implemented.

References

1. Bajaj, C.L., Coyle, E.J., Lin, K.N.: Arbitrary topology shape reconstruction from planner cross sections. Graphical Models & Image Proc. 58, 524–543 (1996)
2. Christiansen, H.N., Sederberg, T.W.: Conversion of complex contour line definitions into polygonal element mosaics. In: Computer Graphics (SIGGRAPH 1978 Proceedings), vol. 12, pp. 187–192 (1978)
3. Park, H., Kim, K., Lee, S.-C.: A Method for Approximate NURBS Curve Compatibility Based on Multiple Curve refitting. Computer Aided Design 32(20), 237–252 (2000)
4. Jha, K.: Energy based multiple refitting for skinning. International Journal of CAD/CAM 5(1), 11–18 (2005)
5. Jha, K.: Catmull-Clark Subdivision and Skinning of Incompatible Curves. In: International Conference on Trends in Product Life Cycle Modelling, Simulation and Synthesis (PLMSS 2006), pp. 83–90. IISc., Bangalore (2006)
6. Nasri, A., Abbas, A., Hasbini, I.: Skinning Catmull-Clark Subdivision Surfaces with Incompatible Curves. In: Proceeding of the 11th Pacific conference on Computer Graphics and applications, PG 2003 (2003)
7. Sederberg, T., Zheng, J., Bakenov, A., Nasri, A.: T-spline and T-NURCUS. ACM Transactions on Graphics, SIGGRAPH 22(3), 477–484 (2003)
8. Doo, D., Sabin, M.: Behaviour of recursive division surfaces near extraordinary points. Computer Aided design 10, 356–360 (1978)
9. Meyers, D., Skinner, S., Sloan, K.: Surfaces from contours. ACM Trans. On Graphics 11, 228–259 (1992)
10. Shinagawa, Y., Kunii, T.L.: The homotopy model: A generalized model for smooth surface generation from cross sectional data. Visual Computer 7, 72–86 (1991)
11. Ekole, A.B., Pyrin, F.C., Odet, C.L.: A triangulation algorithm from arbitrary shaped multiple planner contours. ACM trans. on Graphics 10, 182–199 (1991)
12. Jeong, J., Kim, K., Park, H., Cho, M., Jung, M.: B-Spline surface approximation to cross section using distance maps. Adv. Manuf. Techn. 15, 876–885 (1999)
13. Barequet, G., Shapiro, D., Tal, A.: Multilevel sensitive reconstruction of polyhedral surfaces from parallel slices. Visual Computer 16, 116–133 (2000)
14. Barequet, G., Sharir, M.: Piecewise-linear interpolation between polygonal slices. Comp. Vision & Image Underst. 63, 251–272 (1996)
15. Goodman, T.N.T., Ong, B.H., Unsworth, K.: Reconstruction of C1 closed surfaces with branching. In: Farin, G., Hagen, H., Noltemeier, H. (eds.) Geometric Modelling, pp. 101–115. Springer, London (1993)
16. Bedi, S.: Surface design using functional blending. CAD 24, 505–511 (1992)
17. Wang, H., Qin, K., Tang, K.: Efficient Wavelet Construction with Catmull-Clark Subdivision. Visual Computation 22, 874–884 (2006)
18. Loop, C., Schafer, S.: Approximating Catmull-Clark Subdivision Surfaces by Bicubic Patch, Technical Report, MST-TR-2007-44 (2007)

19. Gabrielides, N.C., Ginnis, A.I., Kaklis, P.D., Karavelas, M.I.: G1-smooth branching surface contruction from cross sections. CAD 39(8), 639–651 (2007)
20. Jha, K.: Construction of branching surface from 2-D contours. International Journal of CAD/CAD 8 (2008)
21. Catmull, E., Clark, J.: Recursive Generated B-Spline Surfaces on Arbitrary Topological Meshes. The Journal of CAD and Application 1(1-4) (1978)

A New Algorithm for Image Resizing Based on Bivariate Rational Interpolation*

Shanshan Gao[1,2], Caiming Zhang[1,2], and Yunfeng Zhang[2]

[1] School of Computer Science and Technology, Shandong University, China
[2] School of Computer Science and Technology, Shandong Economic University, China
gsszxy@yahoo.com.cn

Abstract. A new method for image resizing by bivariate rational interpolation based on function values and partial derivative value is presented. When an original image is resized in an arbitrary ratio, the first step of the method is constructing the rational interpolation fitting the original surface where the given image data points are sampled from. The resized image can be obtained just by re-sampling on the interpolation surface. The algorithm presents how to estimate the partial derivative value of image data point needed for rational interpolation, and at same time considers the adjustment of tangent vector of the edge point to keep edges well defined. Various experiments are presented to show efficiency of the proposed method and that the resized images can preserve clear and sharp borders and hence offer more detail information in real application.

Keywords: Rational Interpolating Spline, Image Resizing, Shape Preserving.

1 Introduction

The problem of resizing images is fundamental and important in the fields such as medical imaging, remote sensing and some software of image processing. This problem arises frequently whenever a user wishes to get better view of a given image, then image resizing methods which can obtain the resized image with higher precision and good quality are required.

Usually, if we can get the interpolating surface fitting the original surface where the given image data points are sampled from, the resized image can be obtained just by re-sampling on the interpolation surface. The interpolation methods can be commonly divided into two types: polynomial interpolation and rational interpolation. At present many polynomial interpolations are popular in image processing application because of their simple implementation. Such as pixel replication or bilinear[1] interpolation. But the visual results of these interpolation methods all suffer from unacceptable effect (e.g. mosaics, aliasing, blocking) to some extent, especially on image edges. In order to

* Project supported by the National Nature Science Foundation of China (No. 60703081, 60673003), the National Nature Science Foundation of Shandong Province (No. Y2007G59), and the Visiting Scholar Foundation for excellent young teachers from colleges of Shandong Province.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 770–779, 2009.

avoid these undesirable defects and obtain good smoothness, a lot of other interpolations are used, such as Bézier[2] or B-spline[3] interpolation, a locally adaptive resizing algorithm[4], resizing method based on vector quantization approximation for magnifying image by a factor of 2[5]. But unfortunately, the results gotten by these methods usually have fuzzy effect in the sharp region, zigzag stripes or blocking effect. Then the visual sharpness of the enlarged image can't match the quality of the original image. This is because most of these methods are based on a simple polynomial model with certain continuity, while preserving the low frequencies content of the source image, but not being able to enhance high frequencies, many details are lost.

However, rational function approximation is a typical non-linear approach, which can reflect the characteristics of the interpolated image surface better, especially it can reflect the mutation between the adjacent image pixels, and hence can keep clearness of edges, describe more complex shapes. References[6-7] use mixed form of polynomial and rational function: using Newton-Thiele as interpolation function to zoom the image and achieved good results. But the interpolation must be used on the whole block with complex computation.

Therefore, in this paper, bivariate rational interpolation model based on function values and partial derivative values is used to fit the original surface of the given discrete image. The zoomed image is obtained just by resampling on the interpolation surface. The partial derivative value of image data point needed for rational interpolation is computed and adjusted to keep shape preserving. In order to maintain the edge property, the adjustment of tangent vector of the edge is presented at same time.

2 Bivariate Rational Interpolation Based on Function Values and Partial Derivative Value

Digital image is a set of color data points on 2D plane. These data are not random, and not complete structural either, because these data are affected by many factors during the acquisition process such as the material of object, light intensity and angle and so on. So the value of adjacent data may have gradual change or have abrupt change. If polynomial is used to interpolate the image data points, it can reflect the gradual change very well because of its form and continuity, but it is inability for the abrupt change. Considering the property of the rational interpolation, we choose bivariate rational interpolation as interpolation function[8], which has the following advantages:1) pass through the known data points; 2) simple and explicit expression; 3) The expression is piece wised, each piece has its parameters, so it is easy for adjusting locally. The concrete form is described as follows:

Let Ω $:[a,b;c,d]$be the plane region, and $\{(x_i, y_i, P_{i,j}, \frac{\partial P_{i,j}}{\partial x}, \frac{\partial P_{i,j}}{\partial y}), i = 1,2,\cdots n; j = 1,2,\cdots m\}$ be a given set of data points, where $a = x_1 < x_2 < \cdots < x_n = b$ and $c = y_1 < y_2 < \cdots < y_n = d$ are the knot spacings, $P_{i,j}, \frac{\partial P_{i,j}}{\partial x}, \frac{\partial P_{i,j}}{\partial y}$ represent $P(x_i, y_j), \frac{\partial P(x,y)}{\partial x}, \frac{\partial P(x,y)}{\partial y}$ at the point (x_i, y_j) respectively. Let $h_i = x_{i+1} - x_i$, $l_j = y_{j+1} - y_j$, and for any point $(x, y) \in [x_i, x_{i+1}; y_j, y_{j+1}]$ in the

(*x*,*y*)-plane, and let $\theta = \frac{x - x_i}{h_i}$ and $\eta = \frac{y - y_j}{l_j}$. First, for each $y = y_j, j = 1,2\cdots m$, construct the x-direct interpolating curve $P^*_{i,j}(x)$ in $[x_i, x_{i+1}]$;this is given by

$$P^*_{i,j}(x) = \frac{p^*_{i,j}(x)}{q^*_{i,j}(x)}, i = 1,2\cdots n-1 \tag{1}$$

Where $p^*_{i,j}(x) = (1-\theta)^3 \alpha^*_{i,j} P_{i,j} + \theta(1-\theta)^2 V^*_{i,j} + \theta^2(1-\theta) W^*_{i,j} + \theta^3 \beta^*_{i,j} P_{i+1,j}$

$$q^*_{i,j}(x) = (1-\theta)\alpha^*_{i,j} + \theta\beta^*_{i,j}$$

and $V^*_{i,j}(x) = (2\alpha^*_{i,j} + \beta^*_{i,j}) P_{i,j} + h_i \alpha^*_{i,j} \frac{\partial P_{i,j}}{\partial x}$,

$$W^*_{i,j}(x) = (\alpha^*_{i,j} + 2\beta^*_{i,j}) P_{i+1,j} - h_i \beta^*_{i,j} \frac{\partial P_{i+1,j}}{\partial x}.$$

with $\alpha^*_{i,j} > 0, \beta^*_{i,j} > 0$.This interpolation is called the rational cubic interpolation based on function values and derivatives which satisfies

$$P^*_{i,j}(x_i) = P_{i,j}, P^*_{i,j}(x_{i+1}) = P_{i+1,j}, P^{*'}_{i,j}(x_i) = \frac{\partial P_{i,j}}{\partial x}, P^{*'}_{i,j}(x_{i+1}) = \frac{\partial P_{i+1,j}}{\partial x}$$

Obviously, the interpolating function $P^*_{i,j}(x)$ on $[x_i, x_{i+1}]$ is unique for the given data $(x_r, P_{r,j}, \frac{\partial P_{r,j}}{\partial x}, r = i, i+1)$ and positive parameters $\alpha^*_{i,j}, \beta^*_{i,j}$.

Using the *x*-direction interpolation function, $P^*_{i,j}(x), i = 1,2,\cdots n-1; j = 1,2,\cdots m$ defines the bivariate rational interpolating function in $[x_1, x_n; y_1, y_m]$ .For each pair(*i*,*j*), $i = 1,2,\cdots n-1$ and $j = 1,2,\cdots m-1$, let $\alpha_{i,j} > 0, \beta_{i,j} > 0$, define the bivariate interpolating function $P_{i,j}(x, y)$ on $(x, y) \in [x_i, x_{i+1}; y_j, y_{j+1}]$ as follows

$$P_{i,j}(x, y) = \frac{p_{i,j}(x, y)}{q_{i,j}(y)}, i = 1,2,\cdots n-1; j = 1,2\cdots m-1 \tag{2}$$

Where

$$p_{i,j}(x, y) = (1-\eta)^3 \alpha_{i,j} P^*_{i,j}(x) + \eta(1-\eta)^2 V_{i,j} + \eta^2(1-\eta) W_{i,j} + \eta^3 \beta_{i,j} P^*_{i,j+1}(x)$$

$$q_{i,j}(y) = (1-\eta)\alpha_{i,j} + \eta\beta_{i,j}$$

and

$$V_{i,j} = (2\alpha_{i,j} + \beta_{i,j}) P^*_{i,j}(x) + l_j \alpha_{i,j} f^*_{i,j}(x, y_j)$$

$$W_{i,j} = (\alpha_{i,j} + 2\beta_{i,j}) P^*_{i,j+1}(x) - l_j \beta_{i,j} f^*_{i,j+1}(x, y_{j+1})$$

with $f^*_{i,s}(x,y_s)=(1-\theta)\frac{\partial P_{i,s}}{\partial y}+\theta\frac{\partial P_{i+1,s}}{\partial y},\theta\in[0,1],s=j,j+1$ .It is obvious that $f^*_{i,s}(x,y_s)$ satisfy $f^*_{i,s}(x,y_s)=\frac{\partial P_{r,s}}{\partial y},r=i,i+1,s=j,j+1\cdot$

The term $P_{i,j}(x,y)$ is called the bivariate rational interpolation based on function values and partial derivative value which satisfies

$$P_{i,j}(x_r,y_s)=P(x_r,y_s),\frac{\partial P_{i,j}(x_r,y_s)}{\partial x}=\frac{\partial P_{r,s}}{\partial x},\frac{\partial P_{i,j}(x_r,y_s)}{\partial y}=\frac{\partial P_{r,s}}{\partial y},r=i,i+1,s=j,j+1$$

It is easy to understand that the interpolating function $P_{i,j}(x,y)$ on $[x_i,x_{i+1};y_j,y_{j+1}]$ is unique for the given data $(x_r,y_s,P_{r,s},\frac{\partial P_{r,s}}{\partial x},\frac{\partial P_{r,s}}{\partial y},r=i,i+1,s=j,j+1)$ and parameters $\alpha^*_{i,j},\beta^*_{i,j},\alpha^*_{i,j+1},\beta^*_{i,j+1},\alpha_{i,j},\beta_{i,j}$.

3 Image Resizing

Suppose that $I_{m,n}$ is a original image with $m\times n$ image data points $P_{i,j}$, $i=0,1,...,m-1,j=0,1,...,n-1$,. These image data are sampling values being regarded as taken from a surface $P(x,y)$. The best way to get a resized image with good quality is to resample from $P(x,y)$. Hence the problem of resizing an original image becomes a problem of constructing fitting surface $P(x,y)$ using the sampling points $P_{i,j}$, $i=0,1,...,m-1,j=0,1,...,n-1$.

Firstly, matrix $P_1=\{P_{i,j},0\le i\le m-1,0\le j\le n-1\}$ is expanded to $P_2=\{P_{i,j},0\le i\le m,0\le j\le n\}$, so the values of $P_{ij}(i=m\ or\ j=n)$ are added. Outer-interpolation method is used, then let $P_{m,j}=2P_{m-1,j}-P_{m-2,j}(0\le j\le n-1)$, $P_{i,n}=2P_{i,n-1}-P_{i,n-2}(0\le i\le m-1)$, $P_{m,n}=P_{m,n-1}+P_{m-1,n}-P_{m-1,n-1}$. At last interpolation surface $P(x,y)$ are constructed based on P_2 .On each interval [x_i,x_{i+1}]×[y_i,y_{i+1}], $i=0,1,...,m-1,j=0,1,...,n-1$, a bivariate rational interpolating spline surface $P_{i,j}(x,y)$ is constructed. To construct $P_{i,j}(x,y)$, see Fig1, the function value $P_{k,l}$, first partial derivatives $(P'_{k,l})_x$ and $(P'_{k,l})_y$ $k=i,i+1,l=j,j+1$, and the parameters $\alpha^*_{i,j},\beta^*_{i,j}$, $\alpha^*_{i,j+1},\beta^*_{i,j+1}$ and $\alpha_{i,j},\beta_{i,j}$ need to be known. The function value

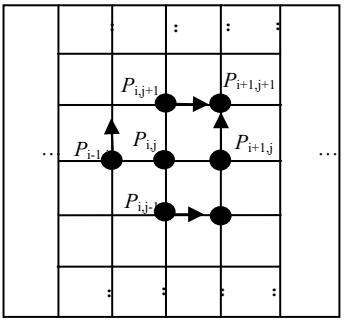


Fig. 1. construction of $P_{ij}(x,y)$

$P_{k,l}$ is the pixel value of original image, $\alpha^*_{i,j}, \beta^*_{i,j}, \alpha^*_{i,j+1}, \beta^*_{i,j+1}$ and $\alpha_{i,j}, \beta_{i,j}$ can be determined according to the condition mentioned above and the practical need. By massive experiments, we find that if $1 \le \alpha_{i,j}, \alpha^*_{i,j}, \alpha^*_{i,j+1} \le 1.5$ and $0.1 \le \beta_{i,j}, \beta^*_{i,j}, \beta^*_{i,j+1} \le 1.0$, the result of the method will be better. Now we discuss the computation of $(P'_{k,l})_x$ and $(P'_{k,l})_y$.

In order to estimate the first order partial derivative $(P'_{i,j})_x$ of $P_{i,j}$, $P_{i-1,j}$, $P_{i,j}$ and $P_{i+1,j}$ are generally used to construct fitting curve in *x*-direction. In similar, points $P_{i,j-1}$, $P_{i,j}$ and $P_{i,j+1}$ can be used to estimate $(P'_{i,j})_y$. Here we take $(P'_{i,j})_x$ as an example to introduce the estimating method for the first order partial derivative of $P_{i,j}$. A quadratic polynomial interpolating curve is constructed by $P_{i-1,j}$, $P_{i,j}$ and $P_{i+1,j}$ and the first order partial derivative is gotten from the curve to approximate $(P'_{i,j})_x$.

$$(P'_{i,j})_x = \frac{\Delta x_i \Delta P_{i-1,j} + \Delta x_{i-1} \Delta P_{i,j}}{\Delta x_{i-1} + \Delta x_i} = \frac{\Delta x_i}{\Delta x_{i-1} + \Delta x_i} \Delta P_{i-1,j} + \frac{\Delta x_{i-1}}{\Delta x_{i-1} + \Delta x_i} \Delta P_{i,j}$$

Where $\Delta x_i = x_{i+1,j} - x_{i,j}, \quad \Delta P_{i,j} = \dfrac{P_{i+1,j} - P_{i,j}}{\Delta x_i}$.

Because digital image can be denote as regular data field, intervals between two neighbor points are equal, so:

$$(P'_{i,j})_x = (P_{i+1,j} - P_{i-1,j})/2 \tag{3}$$

For boundary data points $P_{1,j}$ and $P_{n,j}$, we have

$$P'_i(x_{i-1,j}) = \frac{x_{i-1,j} - x_{i+1,j} + x_{i-1,j} - x_{i,j}}{x_{i-1,j} - x_{i+1,j}} \Delta P_{i-1,j} + \frac{x_{i-1,j} - x_{i,j}}{x_{i+1,j} - x_{i-1,j}} \Delta P_{i,j} = 2\Delta P_{i-1,j} - (P'_{i,j})_x$$

then $(P'_{1,j})_x = 2\Delta P_{1,j} - (P'_{2,j})_x$. In similar: $(P'_{n,j})_x = 2\Delta P_{n-1,j} - (P'_{n-1,j})_x$

Rational spline curve based on function and partial derivative constructed directly with (3) sometimes can't get the shape that given data points advised, which is not acceptable in real application. For example, if the link lines of $P_{i-2,j}$, $P_{i-1,j}$, $P_{i,j}$ and $P_{i+1,j}$ is a planar convex polygon, see Fig2, In order to keep shape preserving, the curve P interpolated to $P_{i,j}$, $P_{i,j}$ $(P'_{i,j})_x$ and Q interpolated to $P_{i-1,j}$, $P_{i,j}$ $(P'_{i-1,j})_x$ must be in the convex region of points $P_{i-1,j}, P_{imid,j}$ and $P_{i,j}$. If P and Q are out of the region we need adjust the derivative computed by (3). We can process as follows:

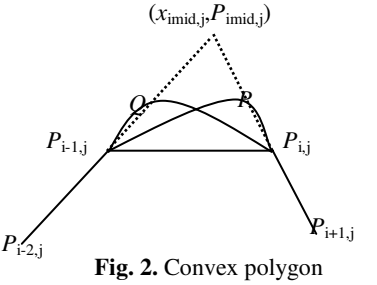


Fig. 2. Convex polygon

$$P = a_1 x^2 + b_1 x + c_1$$

Where $a_1 = \dfrac{P_{i-1,j} - P_{i,j} - (P'_{i,j})_x}{2x_{i,j} - 1}$, $b_1 = (P'_{i,j})_x - 2a_1 x_{i,j} \bullet c_1 = P_{i,j} - a_1 x_{i,j}^2 - b_1 x_{i,j}$

In similar: $Q = a_2 x^2 + b_2 x + c_2$.

The line between $(x_{i,j}, P_{i,j})$ and $(x_{i+1,j}, P_{i+1,j})$ is:

$$P_1 = (P_{i+1,j} - P_{i,j})(x - x_{i,j}) + P_{i,j} = A_1 x + B_1$$

A line between $(x_{i-2,j}, P_{i-2,j})$ and $(x_{i-1,j}, P_{i-1,j})$ is:

$P_2 = (P_{i-1,j} - P_{i-2,j})(x - x_{i-1,j}) + P_{i-1,j} = A_2 x + B_2$ Let $(x_{imid,j}, P_{imid,j})$ denote the intersection point of these two lines, see Fig2. Adjusting $(P'_{i,j})_x$ according to the cross point of interpolation curve P and line P_1. The cross point can be computed by the following formula:

$$P - P_1 = a_1 x^2 + b_1 x + c_1 - (P_{i,j} - P_{i-1,j})(x - x_{i-1,j}) - P_{i-1,j} = 0 \tag{4}$$

Formula (4) has at least one root, namely the interpolating curve and line P_1 will intersect at the end point $(P_{i,j}, x_{i,j})$. If there is one root named x_p which is not end point and satisfies $x_{imid,j} < x_p < x_{i,j}$, it shows that curve P is not in the convex region of points $P_{i-1,j}$, $P_{imid,j}$ and $P_{i,j}$, see Fig2. As well known, if P is tangent with line P_1, P is in the convex region mentioned above. So the shape preserving can be satisfied. Resolving system of equations:

$$\begin{cases} A_1 x + B_1 = a_1 x^2 + b_1 x + c_1 \\ 2a_1 x + b_1 = A_1 \end{cases}$$ then $(P'_{i,j})_x$ is gotten.

In similar, we can get $(P'_{i-1,j})_x$ to make interpolation curve Q tangent with line P_2, and then make Q in the convex region of three points $P_{i-1,j}$, $P_{imid,j}$ and $P_{i,j}$.

During image interpolation process, keeping clearness of edges is necessary. If interpolate with (3), although rational spline can avoid the blurred phenomenon at a certain extent, the edges may be also blur in some case. When a data point P_i is on the edges of digital image, the gray values of its adjacent data points P_{i-1} and P_{i+1} have high difference, we call it a "saltation". Although rational spline can process these "saltation" phenomenon in certain degree, but for some cases that the difference of gray values is comparatively large (such as from white to black), interpolation error will also occur using rational spline. So in this paper, we present an adjusting method for derivative of data points. (3) can determine the derivative of data points. When the gray values of adjacent points have high difference, the derivative of edge points is also high. So we use gradient method to estimate a pixel point is on edge or not. In discrete case, we denote gradient of a pixel as follows:

$$G(x_i, y_i) = \left| df(x_i) \right| + \left| df(y_i) \right|$$

where $\begin{matrix} df(x_i) = f(x_{i+1}) - f(x_i) \\ df(y_i) = f(y_{i+1}) - f(y_i) \end{matrix}$.

In this paper, we give a threshold value to decide a pixel is on edge or not. Threshold value can be taken as $th \in [30,40]$. If the gradient of a pixel is more than given threshold value, we denote it an edge pixel.

Tangent vector can affect the convex degree, so adjusting direction and length of tangent vector on edges to adjust interpolating curve and shape of surface. The blurred phenomenon on the edges of image can be improved greatly. For rational spline interpolation, derivatives of data points are expected to change smoothly. For the edge of image has close relation with the "inner pixels", the derivatives of edge points can be adjusted by weighted of the derivatives of boundary points and the derivatives of points in the around adjacent region. If a pixel point is a boundary point, we use the following formula to adjust the tagent of data points which are on the boundary of image:

$$P_{i,j}{}'(x_i) = \alpha_1 P_{i-1,j}{}'(x_i) + \alpha_2 P_{i,j}{}'(x_i)$$

Where α_1 and α_2 are both tunable parameters. For simplification, denote $\alpha_1 = \alpha_2 = 1/2$.

Then on the interval $[x_i, x_{i+1}] \times [y_i, y_{i+1}]$, the bivariate rational interpolating spline surface $P_{i,j}(x, y)$ based on function value and partial derivative value can be defined as (2). All the surfaces $P_{i,j}(x, y)$ are put together to form the fitting surface $P(x,y)$.

In order to get a resized $m' \times n'$ CT image $I'(x', y')$ from the original $m \times n$ CT image $I(x, y)$, where $m' = m \times s$, $n' = n \times s$, we just need to get more samples by increasing the sample density in accordance with the interval 1/s in x-direction and y-direction of original image respectively to get the image zoomed.

4 Experiments

In this section, the efficiency of the new method is compared with bilinear and bi-cubic interpolation methods, respectively. The new method has been generalized to work with some classical images, such as Lena, Girl, House, GoldHill and peppers and so on, and the results of resizing different part of image Lena with the magnification factor 3*3 and 5*5 respectively, see Fig3 and Fig4, will be taken as examples to show the comparison between the different methods, where image in (a) is original part of image Lena, image in (b) is created by bilinear interpolation to the original image data points, the one in (c) is created by bi-cubic interpolation, the one in (d) is created by the new method. From the images (b)-(d) show that image in (d) has better quality than images in (b)-(c). Fig5 shows another comparison working on the bird image. It is obvious that new method is implemented without producing the so-called mosaics or blocky effect, and the results maintain clearness of the image, including edges, hence offers more detail information. When the image is enlarged by a larger factor, the new method can still present better visual effect.

(a) Part of image Lena

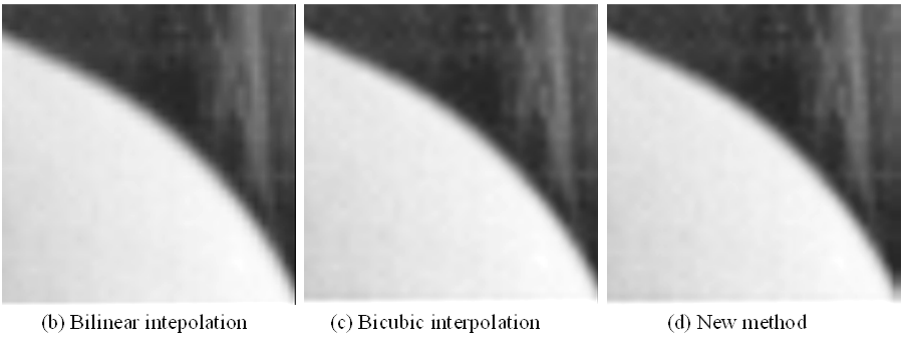
(b) Bilinear interpolation

(c) Bicubic interpolation

(d) New method

Fig. 3. Enlarging for part of image Lena by 3*3

(a) Part of image Lena



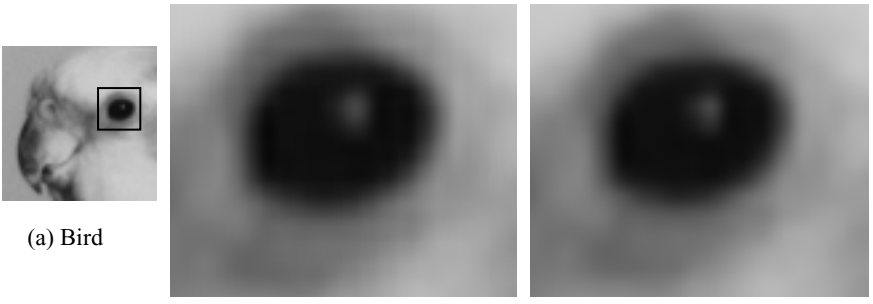
(b) Bilinear intepolation (c) Bicubic interpolation (d) New method

Fig. 4. Enlarging for part of image Lena by 5*5

For a more systematic analysis for the new method, the peak signal-to-noise ratio (PSNR) is used for comparison. PSNR is defined as

$$PSNR = 10log_{10}(\frac{\sum_{ij} 255^2}{\sum_{ij}(I_{ij}^0 - I_{ij})^2})\text{dB}$$

.



(a) Bird (b) Bicubic interpolation (c) New method

Fig. 5. Enlarging for part of image Bird by 10*10

Where $I^0$is the original image and I denotes the recovered image. Firstly, we shrink a selected set of images by 2* 2 using the bilinear interpolation, then magnify by the same factors using the new method and bicubic interpolation respectively. So the original image $I^0$is just the given image, and I is the image gotten after magnifying. Then the PSNR is computed and shown in Table1.

Table 1. PSNR Analysis

Algorithms	Images				
	Girl	Goldhill	House	Lena	peppers
Bicubic	25.472	24.910	23.285	24.503	25.131
New method	27.175	25.673	25.529	26.039	26.871

Besides, we undergo another test. Shrink the selected set of images above by 2* 2, magnify by the same factor, and shrinking and magnifying used the same method. The PSNR is computed and shown in Table2.

Table 2. PSNR Analysis

Algorithms	Images				
	Girl	Goldhill	House	Lena	peppers
Bilinear	25.883	27.311	27.532	28.293	27.367
Bicubic	27.490	27.499	25.038	28.293	26.413
New method	25.560	27.910	27.188	29.866	30.936

As shown in the Tables, the new algorithm improves the PSNR for all cases. We believe that it is due to a denoising feature of the algorithm. The form of rational interpolation, the property of shape preserving and the adjustment on the edgy area are all the reasons for the high quality of the resized image.

The basic algorithm for gray scale pictures can be easily generalized to the case of RGB colored digital images. Three color value R, G, B can be denoted as interpolation

points for rational interpolation respectively, the zoomed image can be obtained by using the interpolation three times.

5 Conclusion

Image resizing by via standard polynomial interpolation methods usually lose more image details, which produces the so-called mosaics or blocky effect, and the edges of zoomed image is blurring. When the magnification factor is bigger, this effect is more evidence. As an important non-linear numerical analysis tool, rational interpolation method can describe the value relation of adjacent pixels better. In this paper, bivariant rational spline function being of advantage to preserving the mutation of the image is used as interpolation function to fit the original surface of the given discrete image. High frequency part of image on the region edge is preserved by adjusting the partial derivative of image data points. Comparing with traditional algorithms, the quality of resized image can be improved greatly with simple computation and arbitrary magnification factor including integer and fractional factor.

References

1. Castleman Kenneth, R.: Digital image processing. Tsinghua University Press, Beijing (1998)
2. Qingjie, S., Xiaopeng, Z., Enhua, W.: A method of image zooming in based on Bézier surface interpolation. Journal of Software 10(6), 570–574 (1999)
3. Zhaoxia, Y., Feng, L.: Guan Lutai1 Image enlargement and reduction with arbitrary accuracy through scaling relation of B-spline. Journal of Computer Aided Design & Computer Graphics 13(9), 824–827 (2001)
4. Battiato, S., Gallo, G., Stanco, F.: A locally adaptive zooming algorithm for digital images. Image and Vision Comput. 20(11), 805–812 (2002)
5. Chang, C., et al.: An image zooming technique based on vector quantization approximation. Image and Vision Computing 23, 1214–1225 (2005)
6. Min, H., Yousheng1, Z.: Image zooming based on Thiele's rational interpolation. Journal of Computer Aided Design &Computer Graphics 15(8), 1004–1007 (2003)
7. Min, H., Jieqing, T., Xiaoping1, L.: Method of image zooming based on bivariate vector valued rational interpolation. Journal of Computer Aided Design & Computer Graphics 16(11), 1496–1500 (2004)
8. Duan, Q., Zhang, H., Twizell, E.H.: A Bivariate Rational Interpolation with Symmetric Bases. Journal of Applied Mathematics and Computation 179(1), 190–199 (2006)
9. Zhang, Y., Duan, Q., Twizell, E.H.: Convexity control of a bivariate rational interpolating spline surfaces. Computers & Graphics 31, 679–687 (2007)

Hardware-Accelerated Sumi-e Painting for 3D Objects

Joo-Hyun Park, Sun-Jeong Kim, Chang-Geun Song[1], and Shin-Jin Kang[2]

[1] Department of Computer Engineering, Hallym University, South Korea
[2] Department of Games, Hongik University, South Korea

Abstract. Brushwork and ink dispersion make it difficult to render 3D common objects in the style of sumi-e painting. We use sphere mapping with brush texture and an image processing techniques to simulate brushstrokes and ink dispersion. The whole process is implemented in shaders running on Graphics Process Unit (GPU) that allows fast and high-quality rendering 3D polygonal models in the style of sumi-e painting. We show several results which demonstrate the practicality and benefits of our system.

1 Introduction

The sumi-e is an East Asian type of brush painting also known as ink and wash painting. Only black ink – the same as used in East Asian calligraphy – is used, in various concentrations. It is non-photorealistic rendering (NPR) which stands in contrast with to conventional graphics rendering methods of photo-realistic. The recent tendency of NPR system is simulating painting style and natural media, e.g. pen and ink, watercolor, charcoal, pastel, hatching, etc. About sumi-e paintings in NPR, many researches of 2D drawing systems have been shown. In these areas, the delicate simulations of brush, black ink and paper are presented, and a 2D image of sumi-e painting is generated accepting the hand drawing of the users.

There have been a number of systems for sumi-e painting brushwork and real-time NPR rendering. Early efforts in sumi-e painting focused on a brushwork simulation. Strassmann[11] swept a one dimension texture to show shading tone. Pham[9] modeled brushstrokes based on variable offset approximation of uniform cubic B-splines. Using the theory of elasticity, Lee[7] modeled a brush as a collection of rods with homogenous elasticity along the entire brush. Way[12] presented a method of synthesizing rock texture in Chinese landscape painting.

With development of GPU technologies, hardware-accelerated rendering skills began to be adapted to NPR system. Kang[4,5] modeled hardware-accelerated rendering algorithm for generating sumi-e painting in real-time from 3D meshes. Chu[2] worked on simulating real-time ink dispersion in absorbent paper using a fluid flow. Yuan[14] developed a GPU-based rendering and animation system for automatically generating Chinese painting cartoon from a set of mesh models.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 780–789, 2009.

In NPR, many systems have addressed real-time NPR rendering. Majumder[8] implemented real-time charcoal rendering applied with contrast enhanced operators by using hardware-accelerated bump mapping and Phong shading. Lake[6] presented a method for cartoon rendering suits for programmable pipeline. Praun[10] introduced tonal art map and showed that it permitted real-time rendering of stroke-based texture for hatching rendering. And he also suggested hardware hatching system with Webb[13] using volume rendering and pixel shading. Kalnins[3] described a way to interactively render stylized silhouettes of animated 3D models with robust frame-to-frame coherence. Chi[1] developed a system for 3D NPR stylized and abstract painterly rendering using a multi-scale segmented sphere hierarchy.

In this paper, we present an interactive system to render 3D objects in the style of sumi-e painting. For burshstrokes we use sphere mapping with brush texture which represents brush style and can be changed for drawing a silhouette in various brush styles. For the interior of 3D models, we shade it using tone texture which can be modified to widen or narrow blank spaces in the painting. To simulate ink dispersion, we use an image processing technique which computes the weighted average of k-texel-away neighbors and then raises to the α-th power. Two parameters k and α are used to control the range and density of spreading. Finally we mix Xuan paper image with the result above to enhance the aesthetic sense. Our system enables users to interactively control all steps by changing brush texture, the parameter of tone texture, the value of spreading range and density, filtering techniques, and Xuan paper image. The whole rendering process is implemented in shaders running on GPU that allows fast and high-quality rendering. Our contributions are the follows:

- We propose two methods to simulate brushstorkes and ink dispersion. One is sphere mapping with brush texture so that brush pattern is mapped on a silhouette. The other is an image processing technique of the filtering controlled by the parameters of spreading range and density.
- We build an interactive rendering system by providing users with as many parameters as possible. As a result, user can draw sumi-e painting as their favors by determining the values of intuitive parameters.

2 Sumi-e Painting Algorithm

Our real-time system has three steps to render 3D models in the style of sumi-e painting: *silhouette outlining*, *interior shading*, and *paper effect*. (Fig. 1) In the first step, for simulating brushstrokes a brush pattern is mapped onto a silhouette of an object using sphere mapping. In the next step, the interior area of the object is shaded based on diffuse reflection and color obtained from object texture. The color of a pixel is determined using the tone texture for sumi-e painting. In the last step, image processing techniques enable to produce paper effect like ink dispersion and mixing background pattern.

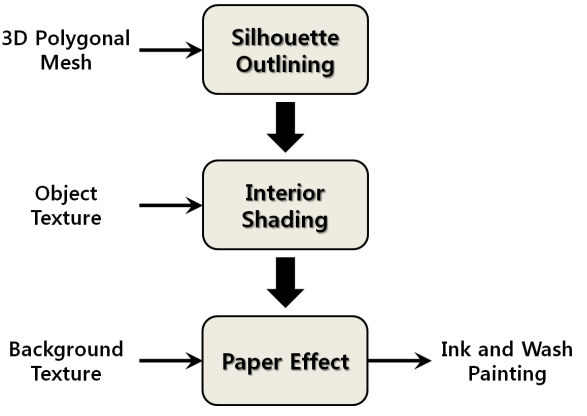


Fig. 1. System overview

2.1 Silhouette Outlining

A silhouette is a view of an object or scene of the outline and a featureless interior. A silhouette edge on a 3D object projected onto a 2D plane (display plane) is the set of points whose outwards surface normal is perpendicular to the view vector. A vertex v is named a *silhouette vertex* if its normal vector $\mathbf{N}_v$ is almost perpendicular to the view vector $\mathbf{V}$.

$$0 \leq \mathbf{N}_v \cdot \mathbf{V} \leq \epsilon$$

where ϵ is the threshold of silhouette extraction and represents the width of silhouette. The value of ϵ can be interactively assigned by users in our system. The larger the number of ϵ is, the thicker the width of silhouette is. If the vertex v is a silhouette vertex then its color is black. Otherwise its color is white.

One of difficulties in rendering 3D objects in the style of sumi-e painting is to render stylized silhouettes with brushstrokes. Unfortunately the scheme above cannot simulate brushstrokes. To achieve brush styles, we use sphere mapping with the brush texture following Kang's approach [4,5]. Sphere mapping can be accelerated by current hardware and has an advantage of not requiring addition calculation for silhouette detection. Also it can show various silhouette drawing effects easily by changing brush texture image.

In view coordinate system, we denote the vector from the vertex to the camera as $\mathbf{v}$, normalized to $\hat{\mathbf{v}}$. Since the computation is performed in view space, the camera is located at the origin and $\mathbf{v}$ is equal to $-\mathbf{p}$, where $\mathbf{p}$ is the position of the vertex in view space. The vertex normal $\mathbf{n}$ is transformed to view coordinates, becoming $\hat{\mathbf{n}}$. The reflected vector $\mathbf{r}(r_x, r_y, r_z)$ can be computed as:

$$\mathbf{r} = 2(\hat{\mathbf{n}} \cdot \hat{\mathbf{v}})\hat{\mathbf{n}} - \hat{\mathbf{v}}$$

We define:

$$m = 2\sqrt{r_x^2 + r_y^2 + (r_z + 1)^2}$$

(a) extracted silhouette (b) brushstrokes (c) brush texture

Fig. 2. A silhouette (a) is changed into a silhouette (b) using sphere mapping with the brush texture (c)

Then the texture coordinates (t_u, t_v) are calculated as:

$$t_u = \frac{r_x}{m} + \frac{1}{2}, \quad t_v = \frac{r_y}{m} + \frac{1}{2} \tag{1}$$

Fig. 2(b) shows the silhouette produced by sphere mapping with the brush texture Fig. 2(c) . As we mentioned before, we can draw silhouettes with various brushstrokes by changing brush texture.

2.2 Interior Shading

Our approach to interior shading is similar to cartoon shading done on the GPU. First, we create a grayscale tone texture that contains the different shade intensities we desire. Fig. 3 shows the tone texture that we use in the rendering system. The tone texture intensity must increase from left to right. To smooth the abrupt transitions between shades, we blur the tone texture using Gaussian function.

Then at each pixel, we perform the standard diffuse calculation dot product to determine the cosine of the angle between the normal vector $\hat{\mathbf{N}}$ and the light vector $\hat{\mathbf{L}}$, which is used to determine how much light the pixel receives:

$$s = \hat{\mathbf{N}} \cdot \hat{\mathbf{L}}$$

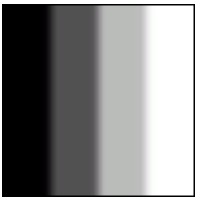


Fig. 3. Tone texture

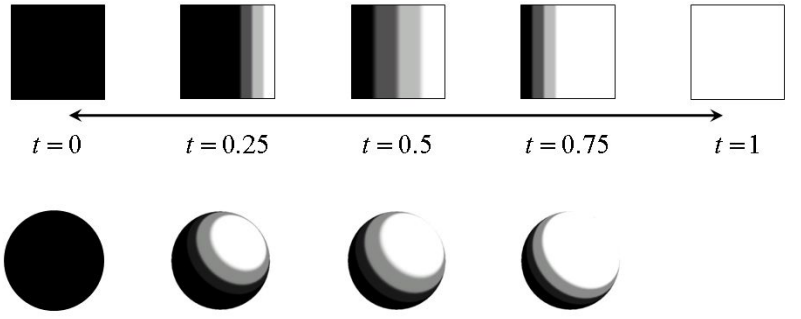


Fig. 4. The width of white shade in the tone texture is determined by the parameter t and simultaneously the region of blank spaces in the painting is determined

If $s < 0$, that implies the surface receives no light. Therefore if $s < 0$, we let $s = 0$. So $s \in [0, 1]$.

Now, we use s to scale our color vector (r, g, b, a) got from object texture so that pixel colors are darken based on the amount of light that they receive:

$$diffuseColor = s(r, g, b, a)$$

Instead of using only s as the u texture coordinate for the shade texture, we compute the luminance of darken pixel color and use it as the u texture coordinate for the tone texture. Therefore the texture coordinates (u, v) are calculated as:

$$u = \min(s(0.3r + 0.59g + 0.11b), 1), \quad v = 0.5 \tag{2}$$

In the sumi-e painting, the concepts of implication and simplicity result in remaining a lot of blank spaces. In other words, many parts of the inside of an object are usually omitted. Our interactive rendering system enables to widen or narrow blank spaces by changing continuously the ratio of white shade to total tone texture. Fig. 4 shows that the parameter t determines the width of white shade in the tone texture. If $t > 0.5$, then the area of white shade is relatively larger than that of black shade. It makes wider blank spaces in the interior of an object than those at $t = 0.5$. Otherwise, the area of white shade is relatively smaller than that of black shade. It also makes narrower blank spaces in the interior of an object than those at $t = 0.5$. If $t = 1$, then all the interior of an object is painted with the white color. In Fig. 4 the bottom row shows the examples of interior shading without the silhouette for a sphere model using the corresponding tone texture.

2.3 Paper Effect

After processes of silhouette outlining and interior shading, the color of a pixel is computed by the multiplication of silhouette and interior colors. We render

not to the screen, but to a texture T which is prepared for image processing and whose resolution is same as that of the rendering window.

$$T(i,j) = Pixel(i,j) = silhouetteColor(i,j) \otimes interiorColor(i,j)$$

where the $\otimes$ symbol denotes component-wise multiplication.

To simulate ink dispersion, we use an image processing technique to compute the weighted average of k-texel-away neighbors from $T(i,j)$ in texture space and then draw it.

$$e(i,j) = \sum_{x=-1}^{1} \sum_{y=-1}^{1} T(i+kx, j+ky) G[x+1][y+1] \tag{3}$$

where k is the range of spreading effect and $G[x][y]$ denotes a 3×3 Gaussian filter. If $k = 1$, then the spreading effect is same as the result image after a 3×3 Gaussian filtering. The larger the value of k becomes, the further the ink is dispersed. In our rendering system, the value of α as well as k can be interactively assigned. The input parameter α plays a role of the density of the spreading effect:

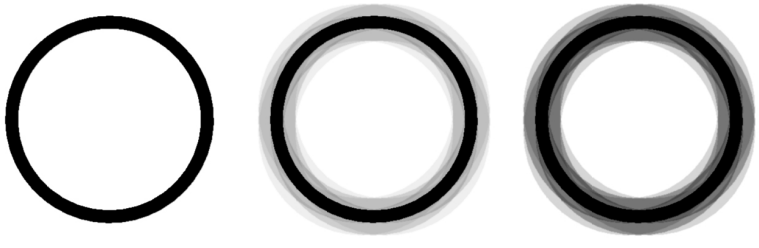
$$filteredColor(i,j) = \mathrm{pow}(e(i,j), \alpha) \tag{4}$$

where the function $\mathrm{pow}(b,n)$ returns b^n. Because $e(i,j) \in [0,1]$, the larger the number of α is, the darker the shade of the spreading effect is. Fig. 5(a) is a simple silhouette of a sphere model without interior shading. Using Equation (3), the silhouette color is spreading k texels away (Fig. 5(b)). If the value of α becomes greater in Equation (4), the spreading effect becomes stronger (Fig. 5(c)).

Finally Xuan paper image is transformed into background texture and mixed with the result above by multiplication.

$$Pixel_{final}(i,j) = T(i,j) \otimes filteredColor(i,j) \otimes backgroundColor(i,j)$$

Because the end result of our rendering system is stored in the texture, we render it onto the screen-aligned billboard whose size is same as that of the rendering window.



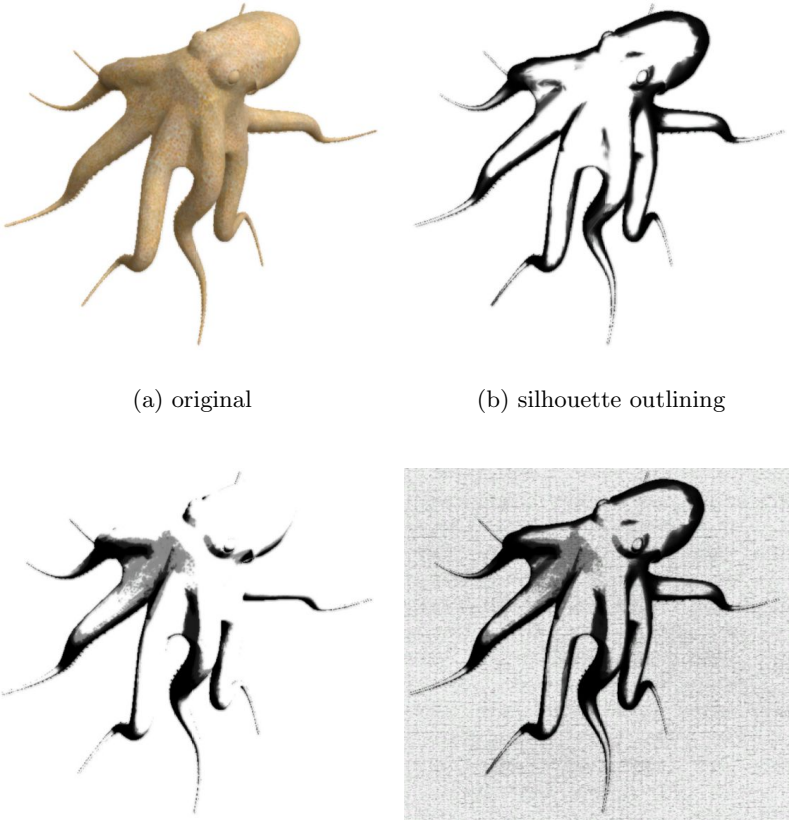
(a) silhouette (b) ink dispersion (c) high density

Fig. 5. Ink dispersion (b) of a silhouette (a) is simply simulated and darkened because of high density (c)

3 Implementation and Results

We have implemented an interactive rendering system using DirectX 9 and HLSL (High-Level Shading Language). The whole process is developed in shaders running on GPU.

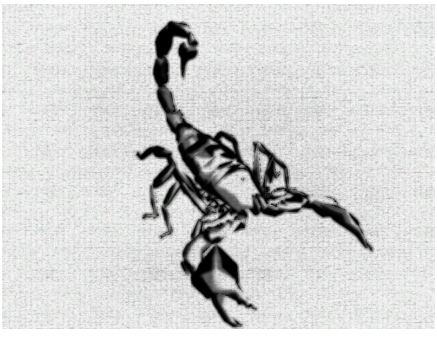
GPU Processing: Our algorithm entirely utilizes programmable GPU vertex and pixel shaders. Because all steps use texture mapping (brush texture, tone texture, and background texture), most operations are implemented in pixel shader. In vertex shader, the position and normal vector of a vertex in world space are transformed into the view coordinates, and texture coordinates pass through it. In pixel shader, silhouette extraction, diffuse reflection, brush texture mapping, interior shading, Gaussian filtering, pulp and spreading effects, and mixing background texture are worked on.



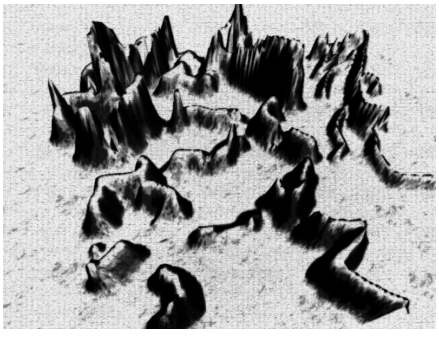
(a) original (b) silhouette outlining

(c) interior shading (d) paper effect + (b) + (c)

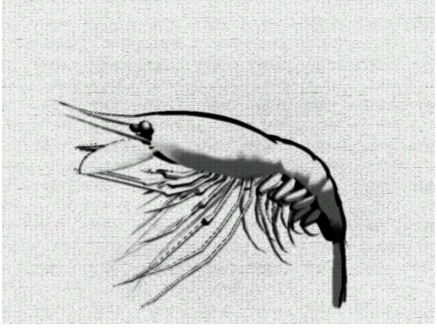
Fig. 6. Sumi-e painting of a octopus model



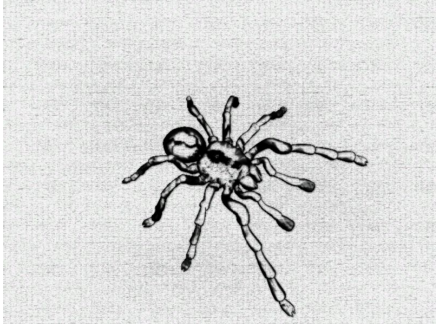
(a) scorpion



(b) terrain



(c) prawn



(d) spider

Fig. 7. Sumi-e paintings generated by our system

Performance: We conducted performance test on a machine with AMD Athlon 64 Dual Core Processor 6000+ CPU and a Geforce 8600GTS GPU with 512MB video memory. The performance data are shown in Table 1. Since half the processing time is consumed in the filtering step, the performance is independent of the number of vertices or triangles. The size of rendering window is the most important factor in the performance. The resolution of rendering window in Table 1 is 800×600. When the resolution is 400×300, the average of rendering performance is $800 \sim 850$ fps. When the resolution is 1920×1200, the average of rendering performance is $100 \sim 150$ fps.

Results: Fig. 2.3 shows an original model and results of each step. Fig. 7 shows sumi-e paintings of various 3D objects. Our input is X mesh files and dds image files for texture. Also we use 3ds MAX as the modeling tool and export models to X files. Users interactively control the brushstroke, shade of interior, spreading effect by filtering technique, and background paper in order to draw sumi-e paintings as their favors.

Table 1. Performance of our system

Object	Vertices	Triangles	Frame/sec.
Octopus	64,014	60,892	452
Scorpion	7,188	10,000	498
Terrain	16,641	32,768	363
Prawn	6,316	7,292	485
Spider	31,467	62,301	454

4 Conclusion

This paper presented an interactive system for rendering 3D objects in the style of sumi-e painting. For burshstrokes, we used sphere mapping with brush texture. We could draw a silhouette in various brush styles by changing this brush texture. For the interior of 3D models, we shaded it using tone texture which was modified to widen or narrow blank spaces in the painting. To simulate ink dispersion, we used an image processing technique which computes the weighted average of k-texel-away neighbors and then raises to the α-th power. Two parameters k and α were used to control the range and density of spreading. Finally we mixed Xuan paper image with the result above to enhance the aesthetic sense. Our system enables users to interactively control all steps by changing brush texture, the parameter of tone texture, the value of spreading range and exponent, filtering techniques, and Xuan paper image. The whole rendering process is implemented in shaders running on GPU.

In future work, we want to apply our rendering system to real-time game like cartoon rendering is already used in games. To do this we try to keep the coherence of silhouette for animating characters and devise rendering method for nature phenomena such as water and smoke in the style of sumi-e painting.

References

1. Chi, M., Lee, T.: Stylized and abstract painterly rendering system using a multiscale segmented sphere hierarchy. IEEE Transactions on Visualization and Computer Graphics 12(1), 61–72 (2006)
2. Chu, N., Tai, C.: MoXi: real-time ink dispersion in absorbent paper. ACM Transactions on Graphics (Proceedings of SIGGRAPH 2005) 24(3), 504–511 (2005)
3. Kalnins, R., Davidson, P., Markosian, L., Finkelstein, A.: Coherent stylized silhouettes. ACM Transactions on Graphics (Proceedings of SIGGRAPH 2003) 22(3), 856–861 (2003)
4. Kang, S.-J., Kim, S.-J., Kim, C.-H.: Hardware-accelerated real-time rendering for 3D sumi-e painting. In: Kumar, V., Gavrilova, M.L., Tan, C.J.K., L'Ecuyer, P. (eds.) ICCSA 2003. LNCS, vol. 2667, pp. 599–608. Springer, Heidelberg (2003)
5. Kang, S.-J., Kim, C.-H.: Real-time 3D sumi-e painting. In: ACM SIGGRAPH 2003 Conference Abstracts and Applications (Technical Sketch) (July 2003)

6. Lake, A., Marshall, C., Harris, M., Blackstein, M.: Stylized rendering techniques for scalable real-time 3D animation. In: Proceedings of NPAR 2000: The 1st International Symposium on Non-Photorealistic Animation and Rendering, June 2000, pp. 13–20 (2000)
7. Lee, J.: Diffusion rendering of black ink paintings using new paper and ink models. Computers & Graphics 25(2), 295–308 (2001)
8. Majumder, A., Gopi, M.: Hardware accelerated real time charcoal rendering. In: Proceedings of NPAR 2002: The 2nd International Symposium on Non-Photorealistic Animation and Rendering, June 2002, pp. 59–66 (2002)
9. Pham, B.: Expressive brush strokes. CVGIP: Graphical Models and Image Processing 53(1), 1–6 (1991)
10. Praun, E., Hoppe, H., Webb, M., Finkelstein, A.: Real-time hatching. In: Proceedings of SIGGRAPH 2001: The 28th Annual Conference on Computer Graphics and Interactive Techniques, August 2001, pp. 581–586 (2001)
11. Strassmann, S.: Hairy brushes. Computer Graphics (Proceedings of SIGGRAPH 1987) 20(4), 225–232 (1986)
12. Way, D., Shih, Z.: The synthesis of rock textures in chinese landscape painting. Computer Graphics Forum (Proceedings of EUROGRAPHICS 2001) 20(3), 123–131 (2001)
13. Webb, M., Praun, E., Finkelstein, A., Hoppe, H.: Fine tone control in hardware hatching. In: Proceedings of NPAR 2002: The 2nd International Symposium on Non-Photorealistic Animation and Rendering, June 2002, pp. 53–58 (2002)
14. Yuan, M., Yang, X., Xiao, S., Ren, Z.: GPU-based rendering and animation for Chinese painting cartoon. In: Proceedings of GI 2007: Graphics Interface, May 2007, pp. 57–61 (2007)

A New Approach for Surface Reconstruction Using Slices

Shamima Yasmin and Abdullah Zawawi Talib

School of Computer Sciences, Universiti Sains Malaysia,
11800 USM Penang, Malaysia
{shamima,azht}@cs.usm.my

Abstract. This paper describes a novel algorithm for surface reconstruction from slices. A number of slices are extracted from a given data oriented along any of the principal axes. Slices are projected onto the XZ plane and equal number of traversals takes place for each slice by a cut plane oriented along the X axis. As the cut plane traverses along each slice, cut points are extracted. To establish correspondence between two consecutive slices, firstly domain mapping takes place. Then a heuristic approach is taken which is based on the comparison of the number of occurrences of particular cut points between slices. Optimization is performed on the basis of minimal differences of the number of occurrences of particular cut points between consecutive slices. Although heuristic approach is not flawless, this algorithm is able to construct surface of fairly complex objects. The algorithm is dynamic enough as the number of slices and the number of traversals can be adjusted depending on the complexity of the object.

Keywords: Contours, Surface Reconstruction, Boundary.

1 Introduction

Contours are usually a sparse representation of an object. Surface reconstruction from slices is performed by taking two consecutive slices into consideration. Surface reconstruction from slices has its application in medical science for surface recovery, for the reconstruction of joint in human body, for volume calculation of human organs etc. In geological field, surface reconstruction is useful to develop terrain model of a particular area from a number of contours. Surface reconstruction has its application in industry for developing synthetic models. Many surface reconstruction algorithms have been developed which construct surface from slices [1-7]. Our proposed algo - rithm was originally developed to construct surface of morphed object [8]. Later it was tested separately for surface reconstruction of a number of objects and found quite sound for constructing surface of fairly complex objects.

2 Related Work

Three dimensional surface reconstruction methods can be classified into the following two categories: (a) Volume-based Surface Reconstruction and (b) Surface Reconstruction

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 790–800, 2009.

from Contours. Volume-based surface reconstruction methods can again be subdivided into three categories: (i) Image Processing Technique, (ii) Distance Field Interpolation (DFI) Method and (iii) Marching Cube Method.

In image processing technique, the whole volume is decomposed into a number of cross-section along a given direction. Slice interpolation or slice replication is performed and these interpolated/ replicated slices are inserted in between the extracted slices [9]. Distance Field Interpolation (DFI) method computes the distance of each voxel within the volume to the surface of the object and interpolates the distance field of the consecutive contours [10]. To take into account the warping of the object, warp guided distance field interpolation is also used [11]. In marching cube method [12], voxel structures within a volume helps to establish the connectivity between the adjacent voxels. Adjacent voxels are connected by a number of triangles which form the constructed surface.

Surface-based reconstruction from contours consists of mapping between adjacent contours. This method consists of the following major pre-processing steps: (i) Discretization of the vertices along the contours, (ii) Finding matched portion of the adjacent contours and (iii) Processing separately of the unmatched portions. Once preprocessing is done, polygonal triangulation is performed. Barequet et al. [5] use some heuristics such as calculation of minimum area, minimum area^2, (minimum area^2 + Length), maximal minimum angle to find the optimum output between the adjacent contour vertices. Sometimes two adjacent contours are converted into a directed toroidal graph whose minimum cost cycle determines the optimal area [2]. Bajaj et al. [3] impose a set of constraints to construct the optimal tiling vertex table between two adjacent contours. To ensure smooth surface, partial differential equation is applied across the contours [6]. Keppel et al. [1] derive optimal triangulated surface for two convex contours from the maximum volume of the polyhedron between them and vice versa for concave contours. Turk et al. [4] construct surface by first converting adjacent slices to two separate implicit functions. Then surface reconstruction takes place between these two functions. Wang et al. [7] at first generate unconstrained delaunay triangulation of all vertices followed by constrained recovery of the boundary edges. Constrained delaunay triangulation is performed for mapping along the surface.

3 Algorithm Overview

Our algorithm is a surface-based reconstruction method. Each contour is first discretized into a number of contour points. Firstly domain mapping between two adjacent contours takes place on the basis of contour overlay. Then on the basis of the number of occurrences of particular cut points, point clouds in each contour are demarcated. Two adjacent contours may differ in the number of cut points. Mapping takes place on the basis of the optimum differences between the number of occurrences of particular cut points in corresponding regions of adjacent slices. After the necessary mapping, rectangular cells are constructed from two pair of points with each pair from each of the adjacent contours.

The algorithm consists of the following two main steps: (a) Contour Pre-processing and (b) Surface Reconstruction. Contour pre-processing step is further subdivided into

three steps: (i) Contour Projection, (ii) Contour Traversal and Extraction of Contour Points and (iii) Orientation and Translation of Contours. Surface reconstruction is also subdivided into three steps: (i) Separation of Disconnected Regions, (ii) Establishing Correspondence between Adjacent Contour Points and Cell Construction and (iii) Consideration of Critical Points.

4 Contour Pre-processing

The detail of this step is as follows:

(i) Contour Projection. All contours are projected onto the XZ plane and centered at the origin.

(ii) Contour Traversal and Extraction of Contour Points. Each contour is traversed along the direction of their minimum X (X_{min}) to maximum X (X_{max}) with a traversal plane defined as (1, 0, 0). Equal number of traversals is performed for each contour. Traversal spacing for each contour is determined as follows:

$$\text{Spacing} = (X_{max} - X_{min}) / \text{Number of Traversals};$$

Boundary points are extracted from the traversals. If the number of extracted points from any cut plane happens to be odd, it is made even. As the cut plane traverses, regions are demarcated. Region is an area where the number of points extracted by the cut plane is the same while traversing along the X axis. In Figure 1(a), the contour consists of 3 regions i.e. two 2-point region and one 4-point region.

(iii) Orientation and Translation of Contours. Each contour defined by a point cloud already projected onto the XZ plane is now oriented along the normal of each contour and translated back to the center of that contour (Figure 1 (b)).

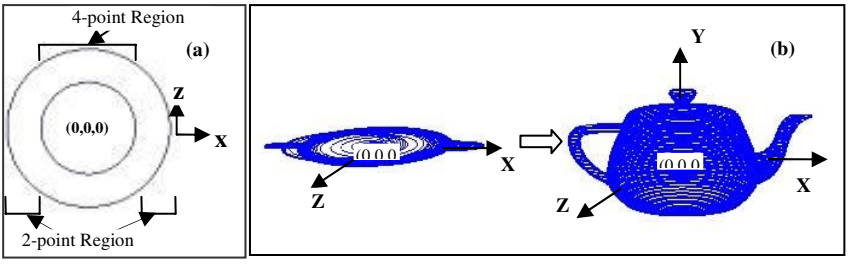


Fig. 1.(a) Region Separation and (b) Orientation and Translation of All Contour Point Clouds

5 Surface Reconstruction

From the stack of oriented and translated point cloud, surface is constructed. Surface reconstruction is performed by only considering each of the two consecutive point clouds. This simplifies the overall surface reconstruction process as where data is highly irregular, necessary modification among cell coordinates is limited to only two consecutive boundaries.

5.1 Separation of Disconnected Regions

Each consecutive boundary may have regions which are disconnected from one another (Figure 2(a)). Nearest neighbor searching is carried out to find out this kind of regions. The disconnected regions are laterally mapped (for better effect) and the other regions are to be mapped across the boundary (vertical mapping), (Figure 2(b)). The detail of vertical mapping is discussed next.

Fig 2. Mapping Separated Regions :(a) Point Clouds with One Region (Top) and Two Regions (Bottom), (b) Vertical Mapping (Left) and Lateral Mapping (Right)

5.2 Establishing Correspondence between Adjacent Contour Points and Cell Construction

After region separation, two consecutive point/ cell arrays (representing two consecu - tive slices) are obtained. Empty region is obtained when the element of the number array (representing number of cut points) is '0'. If the region is empty in one of the arrays while the corresponding element of the other array is not empty (not '0'), then the value '-1' is inserted in the corresponding portion of the other array. By doing so, we obtain equal number of '0' and '-1' regions for both arrays and we extract equal number of sub arrays from these two consecutive number arrays. Now the following operation is carried out on each corresponding extracted sub arrays.

The two arrays which contain the number of interpolated points at each index need to be compressed so that the process of mapping can be carried out in an easier and straightforward manner. Figure 3(a) shows the process of compressing two conse - cutive 'Region Number' arrays (where region numbers are stored) and 'Number of Occurrences' arrays (where the number of occurrences of each region number are stored). Firstly, the size of both arrays should be made equal. This is discussed in detail next.

Each index value of the number of occurrences of the first sub array is compared against the corresponding number of occurrences in the second sub array. To meet the optimum situation, one index value in one of the number of occurrences array may correspond to more than one index values of the other number of occurrences array. Both region number sub arrays and number of occurrences sub arrays are again divided into a number of smaller sub arrays on the basis of corresponding optimum index matching in the number of occurrences arrays. As shown in Figure 3(a), index '0' in the first region number array and the first number of occurrences array correspond to index '0' to index '2' in the second region number array and the second number of occurrences array respectively.

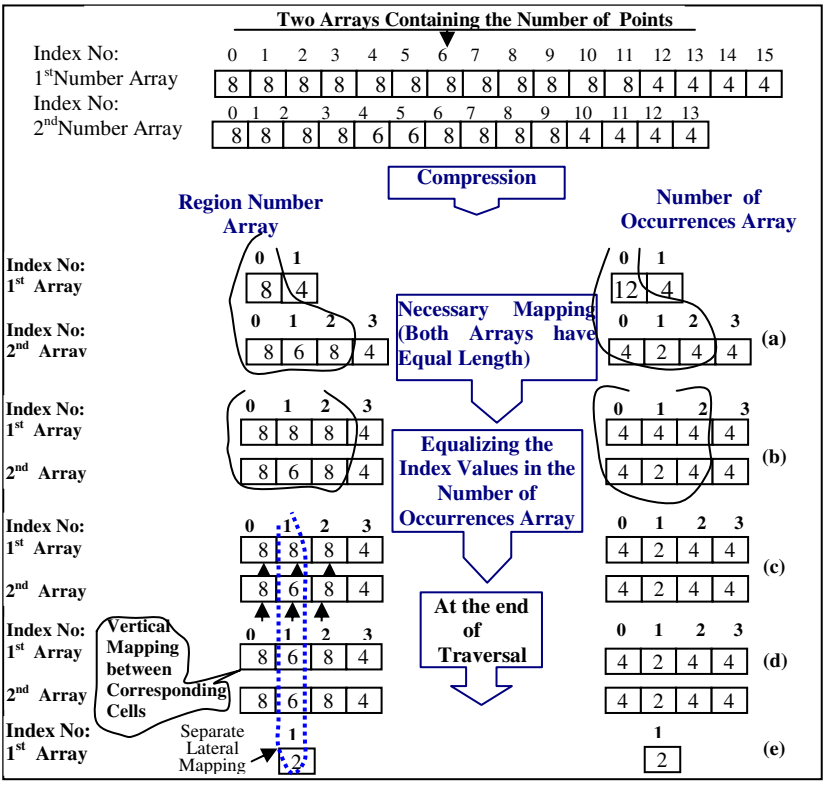


Fig. 3. Establishing Correspondence between Adjacent Contour Points and Cell Construction

After corresponding indices are mapped, the length of both sub arrays may not be equal as shown in the highlighted portion in Figure 3(a). Here the length of the first sub array is '1' whereas the length of the second sub array is '3'. In this case, firstly, the smaller array is to be dissolved to be equal in length of the larger array (Figure 3(b)). Failing which the larger of the two arrays is shrinked to be equal in length of the smaller array. A similar approach is applied for the rest of the array and in the end, both arrays are made to be equal in length. After both 'Region Number' arrays and 'Number of Occurrences' arrays are made equal in length, corresponding values of the number of occurrences arrays are also made equal so that they are ready to be vertically mapped. In the case of unequal values, the larger of the two values is made equal to the smaller number by removing excess number of that particular number of occurrences value (Figure 3(c)).

Corresponding values in the two region number arrays should also be equal for the purpose of vertical mapping. If they are not equal, a further processing needs to be done. The process starts with finding the nearest matched index values of the region number arrays by traversing to the left and the right. The nearest matched values will ensure better continuity between different-numbered regions. Next the corresponding region numbers are split into two portions where the values of the region number of the first portion are derived from the continuous mapping of the nearest matched

index values to the corresponding region number values and the values of the region number of the second portion are the remaining region numbers resulting from the split. In the example (Figure 3(d)), the first discrepancy occurs at index '1' and the nearest matched values are at index '0' and index '2' with a value of '8' and '8'. The current value i.e. 8 in the first array needs to be split into two portions. The first portion is made equal to '6' and the second portion is assigned the remaining value (8-6 = 2). At the end of the entire processing, two sets of region number arrays are obtained. The top set (Figure 3(d)) now consists of equal region number and can therefore be vertically mapped whereas the bottom set (Figure 3(e)) is to be laterally mapped separately.

5.3 Consideration of Critical Points

5.3.1 Empty Space Consideration

When there is an empty space in both corresponding cells, spaces are separated by direct vertical mapping in between them (Figure 4(a)). On the other hand, when solid cells in one boundary meet empty space in another boundary during vertical mapping, corresponding imaginary empty space is calculated for the solid portion. Now empty spaces are separated by direct vertical mapping and lateral cell construction takes place for the imaginary empty portion (Figure 4(b)).

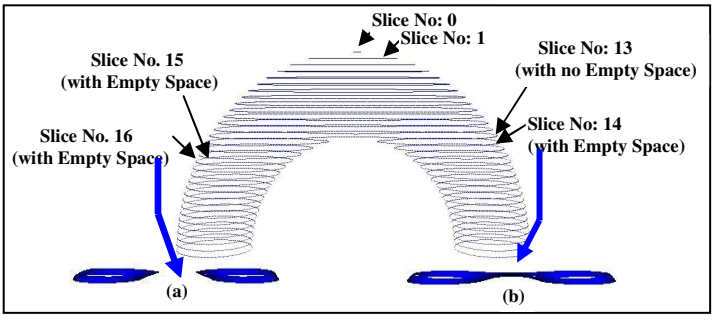


Fig. 4. Empty Space Consideration during Surface Reconstruction: (a) Reconstructed Surface from Slice No. 15 and Slice No. 16 and (b) Reconstructed Surface from Slice No. 13 and Slice No. 14

5.3.2 Transition between Different-Numbered Regions

Discontinuity occurs at the transition point between two different-numbered regions. To ensure smooth transition between the regions, the last point array of the first region and the first point array of the second region have to be scanned so that the portions which are continuous and the portions which are not continuous are identified (Figure 5).

5.3.3 Distinguishing Connected/ Disconnected Regions across the Direction of Traversal of Slices

When region number values in both corresponding region number arrays are equal and region number value is greater than two, and there exists an empty region next to

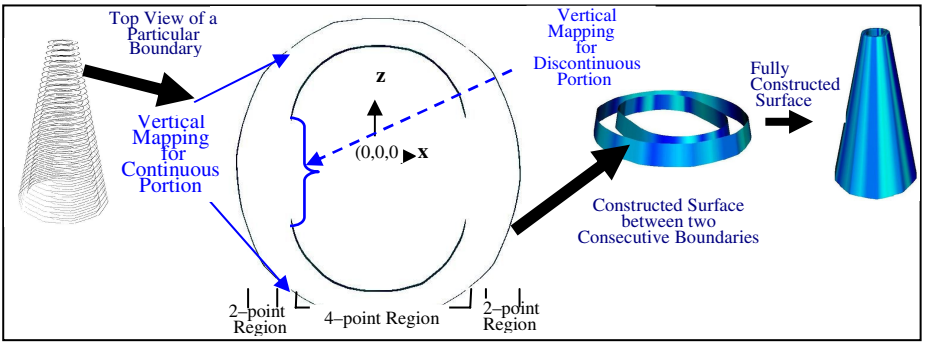


Fig. 5. Surface Reconstruction while Transiting between Different-numbered Regions

it, there should also exist empty space across the section. However there may be ambiguity on the number of empty spaces across the section. Figure 6 shows two different cases of two adjacent point clouds where the region number in the point clouds is '4'. In Figure 6(a), there are two empty spaces across the section whereas in Figure 6(b), there is one empty space across the section.

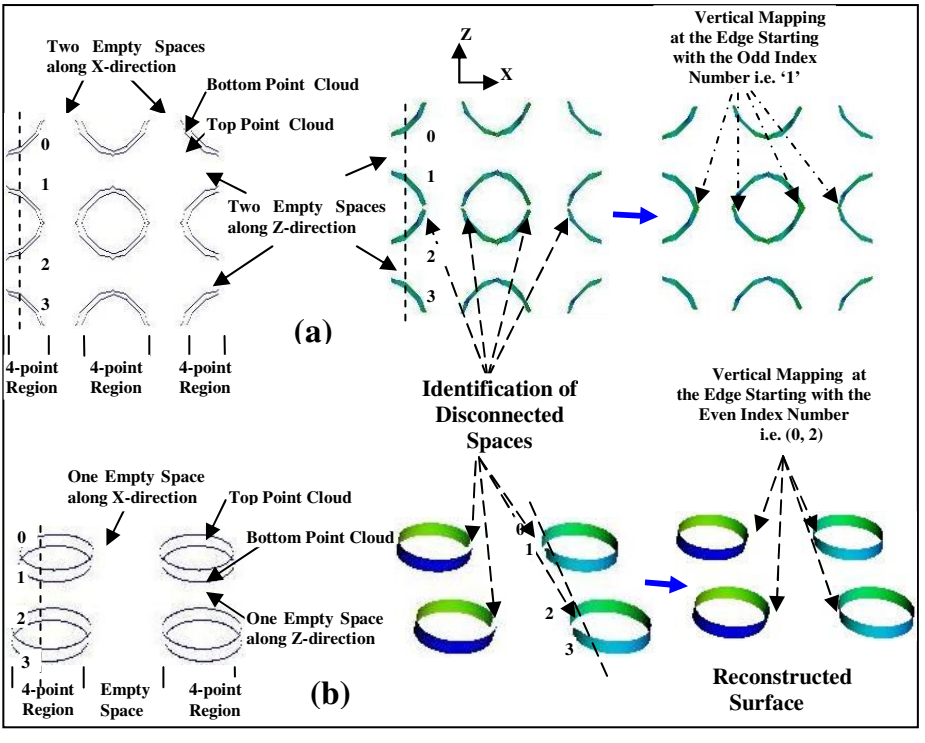


Fig. 6. Surface Reconstruction Considering the Number of Empty Spaces across the Slice: (a) Two Empty Spaces and (b) One Empty Space

In addition to these cases, it may happen that only one of the two consecutive slices has empty spaces. In this case the same method as mentioned above is applied and in addition, lateral mapping takes place for the non-empty portion.

6 Implementation and Results

The algorithm is implemented using C++ with Visualization Tool Kit (VTK) as graphics platform. WindowsXP is used as the operating system with the following hardware configurations: Pentium 4-M CPU 2.40GHZ, 1 GB of RAM. In Figure 7(a) and Figure 7(b), surfaces are reconstructed from 129 slices with the number of traver -sals 201 with a runtime of 94 seconds and 161 seconds respectively. In Figure 7(c) and Figure 7(d) two parametric surfaces i.e. "crosscap" and "boy" are constructed from 129 slices with the number of traversals of 201 and with the runtime of 84 seconds and 88 seconds respectively. In Figure 8(a) and Figure 8(b) two surfaces i.e. "Heart" and "Human head" are reconstructed from 129 slices with the number of traversals 201 and 401 respectively and a runtime of 97 seconds and 200 seconds respectively.

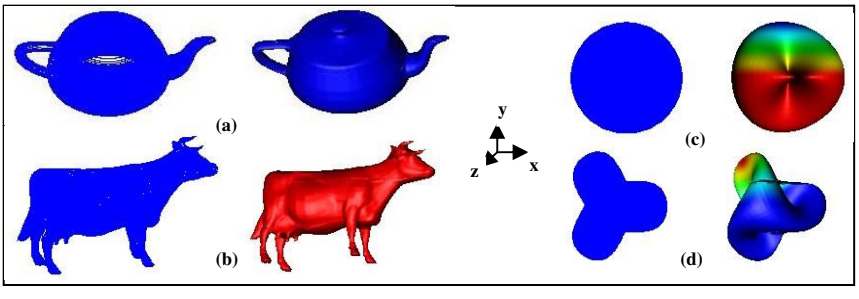


Fig. 7. Reconstruction of Surface from Slices Arranged along the Y-axis

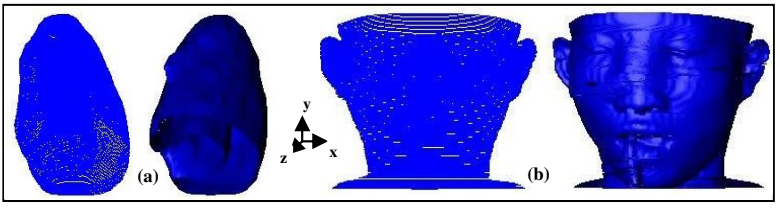


Fig. 8. Reconstruction of Two Surfaces from Slices Arranged along (a) the Z-axis and (b) the Y-axis respectively

Figure 9 shows how the output as well as runtime vary as the number of slices and the number of traversals vary. Figure 9(a)(i) shows the output for a parametric surface "Figure-8 Klein" with a runtime of 10 seconds with the number of slices 33 and the number of traversals 101. In Figure 9(a)(ii), the same parametric surface is reconstruct -ted from 65 slices with the number of traversals 201 and a runtime of 39 seconds whereas in Figure 9(a)(iii) runtime is 166 seconds when the number of slices and the

number of traversals are 129 and 401 respectively. Runtime increases as the number of traversals as well as the number of slices increase as shown in the Figure 9(b).

As it is discussed in Section 5.3.3, slices may also need to be traversed along Z-axis in order to find out the number of empty spaces along that direction when there is ambiguity about the number of empty spaces along that direction. In Figure 10, a surface called "schwarz" is reconstructed from 129 slices with the number of traver-sals 201

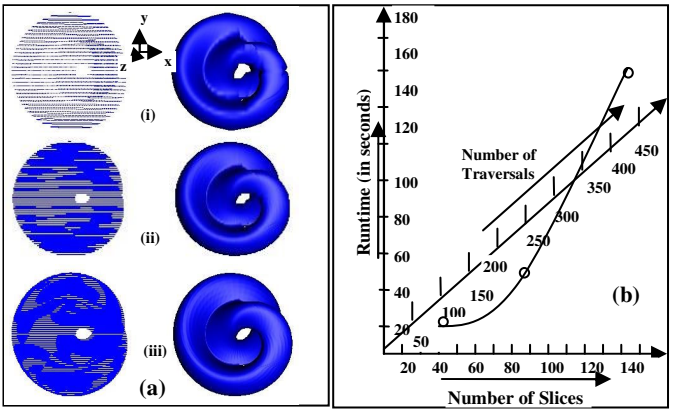


Fig. 9. Variation in (a) the Output and (b) Runtime with the Variation in the Number of Traversals and the Number of Slices

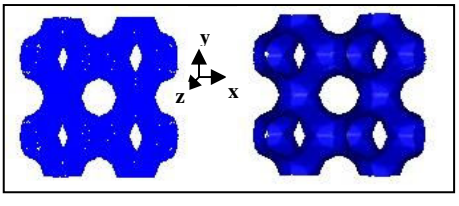


Fig. 10. Reconstruction of a Surface from Slices Arranged along the Z-axis

Table 1. Variation in the Number of Cells and Runtime with Variation in the Number of Slices and the Number of Traversals

Name of Object	Number of Slices	Number of Traversal along Each Slice	Direction of Arrangement of Slices	Number of Cells	Runtime (in seconds)
(a) Human Head	129	401	Y-axis	124739	200
(b) Heart	129	201	Z-axis	71805	97
(c) Cow	129	201	Y-axis	42928	161
(d) Teapot	129	201	Y-axis	49044	94
(e) Parametric Surface (crosscap)	129	201	Y-axis	52000	84
(f) Parametric Surface (boy)	129	201	Y-axis	62170	88
(g) Schwarz	129	201	Z-axis	19859	340
(h) Parametric Surface (Figure-8 Klein)	33	101	Y-axis	7108	10
	65	201	Y-axis	28238	39
	129	401	Y-axis	112547	166

and a runtime of 340 seconds. There is an increase in the runtime because of this excess of traversing along the Z-axis. Table 1 summarizes the variation in the runtime and the number of cells with the variation in the number of traversals and the number of slices for the different objects mentioned above.

Let us analyze the algorithm in terms of efficiency. Let us suppose that 'n' is the number of traversals along the X-axis, 'm' is the number of traversals along the Z-axis, 'p' is the tuple number of 'Region Number Array' in two consecutive slices. During contour point extraction, each slice is traversed along the X-axis. During tuple equalization, comparison of each index value of 'Region Number Array' takes place. During vertical mapping, rightward and leftward traversals take place for each slice which in worst case can be equal to the tuple number in the 'Region Number Array'. Again traversal along the Z-axis can take place for each slice during vertical mapping. Considering the number of constructed cells for each of the two consecutive slices is equal to the number of traversal along the X-axis i.e. 'n', the complexity per slice is $\Theta((n + p + pm) + n)$. Here 'p' signifies the tuple number of the 'Region Number Array' and its value is usually much smaller than 'n' and 'm'. Hence the complexity is reduced to $\approx \Theta(n + pm)$.

7 Conclusion and Future Work

The proposed algorithm shows a simple method of constructing surface from slices. Some compromises have been made in developing this algorithm as it is already men -tioned that this algorithm was originally developed to reconstruct surface for morph - ed objects where accuracy is not so stringent. Still this method works well as a stand - alone algorithm for reconstruction of surfaces ranging from simple to fairly complex objects. No triangulation is performed in this method. Depending on the complexity of the object, the number of slices and the number of traversals can be adjusted and the runtime can also be varied. As surface reconstruction is limited between two adja - cent slices, discontinuities may be noticeable near the edges. This method also works well when slices are not parallel. Future works involve strengthening the algorithm so that it works well as a full fledged standalone algorithm for surface reconstruction.

References

1. Keppel, E.: Aproximating Complex Surface by Triangulation of Contour Lines. IBM Journal of Research and Development 19, 2–11 (1975)
2. Fuchs, H., Kedem, Z., Uselton, S.: Optimal Surface Reconstruction from Planar Contours. Communications of the ACM 20(10), 693–702 (1977)
3. Bajaj, C., Coyle, E., Lin, K.: Arbitrary Topology Shape Reconstruction from Planar Cross Sections. Graphical Models and Image Processing 58, 524–543 (1996)
4. Turk, G., O'Brien, J.: Shape Transformation Using Variational Implicit Functions. In: Proc. SIGGRAPH, pp. 335–342 (1999)
5. Barequet, G., Shapiro, D., Tal, A.: Multi-level Sensitive Reconstruction of Polyhedral Surfaces from Parallel Slices. The Visual Computer 16(2), 116–133 (2000)
6. Hormann, K., Spinello, S., Schröder, P.: C-1-continuous Terrain Reconstruction from Sparse Contours. In: Proc. Vision, Modeling and Visualization, pp. 289–297 (2003)

7. Wang, D., Hassan, O., Morgan, K., Nigel, W.: Efficient Surface Reconstruction from Contours Based on Two-Dimensional Delaunay Triangulation. International Journal for Numerical Methods in Engineering 65, 734–751 (2006)
8. Yasmin, S., Talib, A.Z.: A Method for 3D Morphing Using Slices. In: Proceedings of International Conference on Computer Graphics Theory and Applications (GRAPP), pp. 292–301 (2009)
9. Levoy, M.: Display of Surface from Volume Data. IEEE Computer Graphics and Applications 8, 29–37 (1988)
10. Payne, B., Toga, A.: Distance Field Manipulation of Surface Models. IEEE Computer Graphics and Applications 12(1), 65–71 (1992)
11. Cohen-Or, D., Levin, D.: Guided Multi-Dimensional Reconstruction from Cross-sections. In: Fontanella, F., Jetter, K., Laurent, P.J. (eds.) Advanced Topics in Multivariate Approximation, pp. 1–9. World Scientific Publishing Co., Singapore (1996)
12. Lorensen, W., Cline, H.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm. Computer Graphics 21, 163–169 (1987)

Tools for Procedural Generation of Plants in Virtual Scenes

Armando de la Re, Francisco Abad, Emilio Camahort, and M.C. Juan

Depto. Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
46022 Valencia, Spain

Abstract. Creating interactive graphics applications that present to the user realistic natural scenes is very difficult. Natural phenomena are very complex and detailed to model, and using traditional modeling techniques takes huge amounts of time and requires skilled artists to obtain good results.

Procedural techniques allow to generate complex objects by defining a set of rules and selecting certain parameters. This allows to speed up the process of content creation and also allows to create objects on-the-fly, when needed. On-demand generation of scenes enables the authors to create potentially infinite worlds.

This survey identifies the main features of the most used systems that implement procedural techniques to model plants and natural phenomena and discuss usability issues.

1 Introduction

Massive multiplayer games model huge environments where lots of players can exchange experiences. To obtain appealing landscapes and interesting game fields, modern games require lots of both geometric and texture assets. This pose a difficult problem since it is very expensive to create lots of different objects, landscapes, characters and so on. It is common to reuse such content in the same game. Changing certain characteristics of the object (i.e., its color, its size) to increase the number of different objects in the game is usually detected by the user, thus reducing the realism of the game.

The resources dedicated to create realistic models could be used to improve game play or include innovative features. Procedural content generation techniques appear to speed up the process of creating content. They are also able to generate content on-the-fly, thus reducing the space requirements.

Recently, automatic content creation systems have been used, for example, to model buildings and cities [1,2], roads [3], buildings [4,5], houses [6], textures [7], vegetation [8,9] and sky [10]. Specifically for games, procedural systems have also been used to model 2D maps [11] and game levels [12].

Usually one of the requirements of games is to present realistic vegetation. This is a difficult goal because natural plants are complex organisms and different factors define its shape and color. It is possible to model a realistic plant

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 801–810, 2009.

with traditional methods, but it usually results in a huge geometric model, with lots of textures, and it is a time consuming task. Rendering complex models also requires applying some technique of LOD to reduce the actual number of polygons processed in the scene. Some procedural generation algorithms are able to generate multi resolution models [9] and some others are able to generate plants based on images [13].

This work focuses on currently available software applications that use procedural generation algorithms to model plants. We describe the features of the most used applications, and we also study their usability.

The rest of the article is structured as follows. First we talk about previous work on procedural content generation. The following section describes the main features of each surveyed system. We provide a table that compares each aspect of the applications. Conclusions and future work ends the paper.

2 Previous Work

The first procedural techniques were based on recursive functions, and were used to create fractal-like images. A fractal is a fragmented geometry shape, where each fragments is (approximately) a reduced copy of the whole shape (self-similarity). They cannot be classified in the traditional Euclidean geometric system, have a rich structure at arbitrarily small scales and have a Hausdorff dimension greater than its topological dimension. Some generation techniques use fractals to generate plants, rocks [14] and other natural phenomena [15].

Other procedural techniques are based on L-Systems [16]. L-Systems are a variation of formal grammars and are used to simulate the growth of plants, fractals and artificial life. They consist of a set of symbols that can be replaced, an axiom or initial state and a set of production rules. The L-System starts with the axiom that is replaced with the corresponding production rule. Then some parts of the rules are replaced with other rules and so on. The results are interpreted by the renderer as positions, orientations and stack structures. This kind of procedural technique is used to generate complex plants and other natural structures.

Many authors have focused on procedural techniques for city generation [17]. We can find applications to generate cities in a terrain [1], organize and simulate cities with procedural methods [2], create roads and streets [3], green areas, bridges, etc. Other systems model buildings [4], houses [6], facades [5]... These systems have to create different buildings, but have to maintain a common look (for example, to model different buildings built around the same era with the same style). Kelly and McCabe [17] defined seven criteria to evaluate a procedural city generation system: realism, scale, variation of buildings and roads, required input to generate it, efficiency, control to modify the generation, and if it is generated in real time.

Other authors have presented techniques for modeling realistic trees [8], reducing the polygon count of the plant model using procedural techniques [9], image-based modeling of plants [13] or animating plants [18].

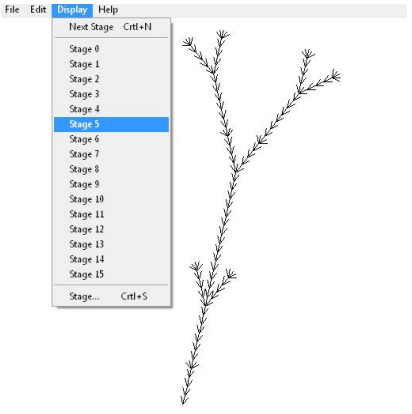


Fig. 1. FractTree allows to render separately each stage of the derivation

Fig. 2. The preview and navigation screen of L-System4

3 Applications

3.1 FractTree [19]

This application is one of the precursors in the generation of fractal plants. It creates only 2D models, and uses L-Systems and a step by step generation with detail level for derivation rules shown in Figure 1. The application creates the plant replacing the symbols in the derivation with drawing primitives. It is a very simple program but it can be used to understand the basics of L-Systems.

3.2 L-System4 [20]

It is also based on L-Systems, and generates detailed 3D plants and objects (see Figure 2). The navigation is somewhat restricted but it is enough to examine the object. One problem of this application is that the user has to know how L-Systems work to create or change one.

3.3 LStudio [21]

LStudio provides several tools to create realistic plants as shown in Figure 3.a. It is based on a modified bracketed L-System to generate trunks, branches and the position of leaves, flowers and petals. These terminals are modeled in the interactive vector editor shown in Figure 3.b. This system is suitable to generate small plants like flowers, grass and bushes, rather than trees.

3.4 An Ivy Generator [22]

It is a generator of Ivy plants that allows the user to decide where to grow them on an imported 3D scene by defining a seed. It has simple tools to change the

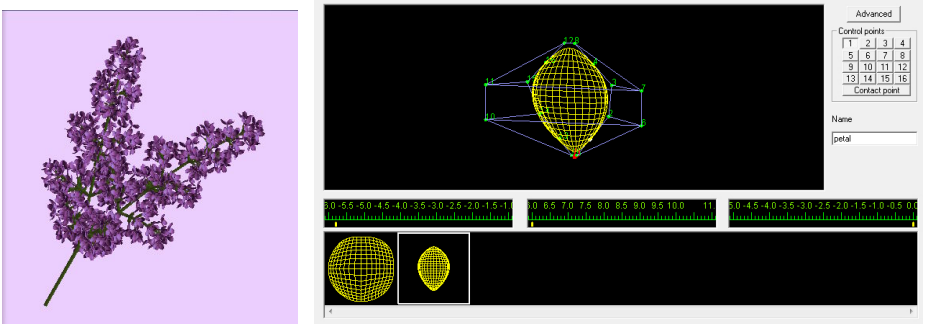


Fig. 3. a) Plant created with LStudio, and b) its editor for modeling leaves, flowers and petals

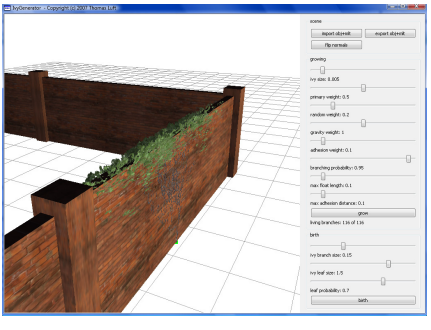


Fig. 4. An Ivy Generator example

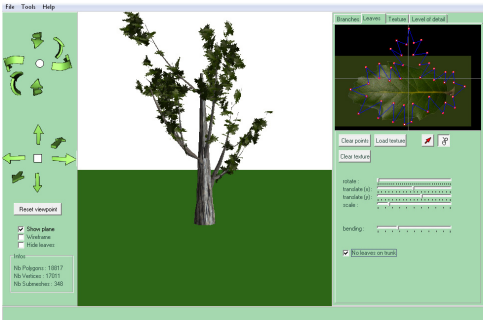


Fig. 5. Treegenerator GUI

appearance of the plants but the results are very realistic. It takes into account the gravity and the capacity of the plant to grow to create climbing or hanging plants (Figure 4).

3.5 TreeGenerator [23]

This application has a control panel to control the tree generation. Figure 5 shows the leaf editor. The resulting leaves look real when isolated, but groups of leaves do not look realistic. One of the causes is that the program has a limited number of recursion levels to generate branches and leaves. The tools to modify the tree and create different instances are also limited.

3.6 TreeMagik G3 [24]

This tool provides a trunks, branches and roots generator. The foliage is provided by the program and it is rendered as a set of billboards. It generates very good results as shown in Figure 6. It is also able to generate a billboard of the entire tree. It provide textures for the trunk and leaves, and the user can add textures.

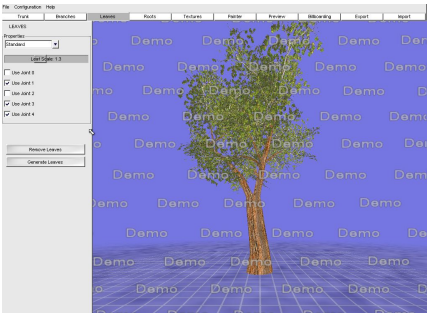


Fig. 6. Example of a tree generated by TreeMagikG3

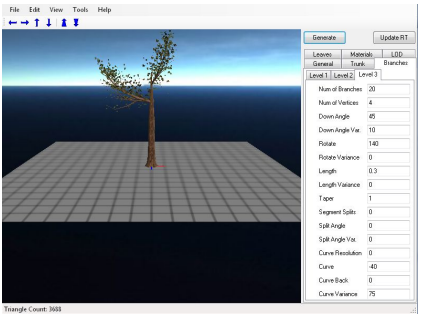


Fig. 7. Example generated with Meshtree Studio

Fig. 8. Dryad generates trees quickly, but they are not very realistic

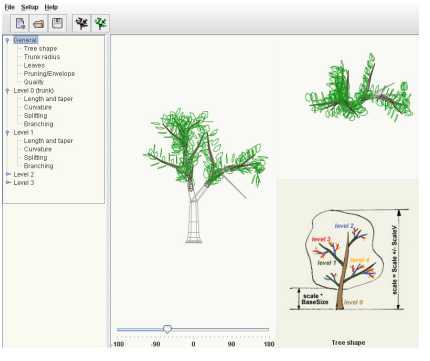


Fig. 9. The help system of Arbaro provides information about the parameters to control the generation

3.7 MeshTree Studio [25]

It produces very realistic trees but the user interface is not very user friendly and there is no help or tutorial. Once the initial learning curve is overcome, the user can generate very appealing trees with a low number of polygons. One restriction is that only generate *.mesh* files.

3.8 Dryad [26]

This is a freeware tree generator, but it is not open source. It provides an online gallery of trees that looks like a forest, where the user can select a tree and change its parameters. The properties of two different trees can be combined to create a new tree. The trees created by the users can be planted in the online gallery and shared with other users. A disadvantage of this system is that it only generates high resolution trees and they are not very realistic (Figure 8).

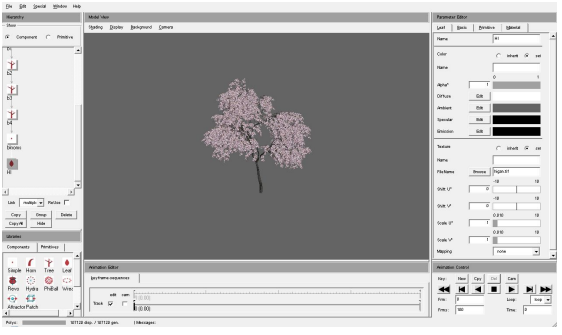


Fig. 10. Xfrog Graphical User Interface

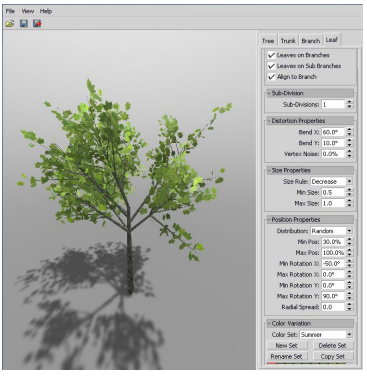


Fig. 11. Tree[d] produces good results but it is difficult to control variations

3.9 Arbaro [27]

This is the only Java-based system evaluated and generates good results. It is well documented, but its interface is not very friendly. It has some errors exporting trees. Figure 9 shows the flexibility to control parameters like number of levels, radius, division and curvatures of branches and trunk. It provides overlay graphics aids to help the user to understand each parameter.

3.10 SpeedTree [28]

It's one of the most used and renown products in the creation of videogames that present natural scenes. It provides a powerful, complete, and efficient renderer, a modeler and real time generator. The scenes created with this engine are very realistic, and it has also been designed to be used in interactive systems. This is the most expensive system described in this survey, and we could not test it.

3.11 Xfrog [29]

It is similar to SpeedTree and it is also used in the videogame industry. It can generate very realistic scenes, with different objects and a wide variety of plants. It is more affordable than SpeedTree. Its learning curve is also very steep, and takes time to obtain a convincing tree.

3.12 Tree[d] [30]

It is very easy to create random trees and generates very realistic examples. It is difficult to modify or start a new one, and there is not many different type of trees.

4 Conclusions and Future Work

The purpose of this work was to study the current procedural generation systems to model natural plants. We have classified this tools into several profiles, depending on the features of the systems and the requirements of the user: FractTree and L-System4 is suitable for students, since they help to learn and understand L-Systems. Speedtree and Xfrog are suitable for game and movie companies because they generate the most realistic trees, but at the expense of a steep learning curve and a high cost. L-Studio also creates realistic flower plants but it requires a lot of learning time. An Ivy Generator makes a scene or object look abandoned or alive because of the added Ivy plant, and because it is free, it can be used by everyone. Treemagik G3 and Tree[d] also generate good results and are cheap or free.

Some systems can be used by 3D cartoon animators to generate trees. Dryad, TreeGenerator and L-System4 can generate cartooned trees instead of realistic plants. For the casual user FractTree, dryad, MeshTree studio, TreeGenerator and An Ivy generator provide a lot of parameters to select and different results.

Our group is currently developing a framework for procedural modeling of synthetic models. This tool will allow the user to select different techniques to generate geometry and textures inside the same environment.

Summary

This section includes a table comparing the main features of each system. The features studied are:

- 2D and 3D: Capacity to generate 2D and/or 3D objects.
- IDE: Integrated editor to create the plants.
- Definition Language: Input for plant definition. It can be LSys [] (bracketed L-Systems) or GUI (Graphical User Interface).
- Navigation: Navigation in the rendered scene (Zoom, move, rotation).
- Geometry Generation: It can generate geometry.
- Import, Export: file formats accepted by the application.
- Released, Updated: First release and last update dates.
- Derivation Control: Control of the derivation of plants.
- Multiple Objects: Render multiple objects at the same time.
- Object types: Objects that the system can work with.
- Textures: Capacity to use textures in plant creation.
- Purpose: Main purpose of the application.
- Usability: Easy of use in a 1-10 scale (1 difficult, 10 easy).
- Documentation: Quality of the documentation in a 0-10 scale (0 no documentation, 1 poor, 10 complete and user friendly).
- Debugging: Tools for debugging the generation process.
- Modeling Speed: Time to create a target plant in each system. It could be from a blank project or modifying an existing example. The plant has these requirements: An initial branch (or trunk) and a separation in two branches, five derivations and medium-sized round leaves.

Table 1. Main characteristics of the surveyed applications. Legend for "Modeling Speed" row: : 1, fast using examples and knowing L-Systems; 2, very slow, required modifying existing examples; 3, very fast, requires an external object; 4, fast to build but difficult to modify; 5, very fast; 6, fast but export does not work; 7, normal and 8, we could not test it

	FracTree	L-System4	L-Studio Botany Alg	An Ivy Generator	TreeGenerator	TreeMagik G3
2D	x	-	x	-	-	-
3D	-	x	x	x	x	x
IDE	x	x	x	x	x	x
Definition Language	Lsys □	Lsys □	Mod. Lsys □	GUI	GUI	GUI
Navigation	-	x	x	x	x	x
Geometry Generation	-	x	x	x	x	x
Image Generation	-	x	x	x	x	x
Import	-	dxf	-	obj	-	b3d
Export	bmp	dxf, bmp, jpeg	rgb, tga, bmp	obj	obj, 3ds, dxf	several (obj, wrl)
License	Shareware	Freeware	Freeware	Freeware	1.3 & 2.0	Demo
Price	10 EUR	0	0	0	$0 & $49	$49.95
Released	1993	2000	1999	2007	2006	2006
Updated	No	2004	2004	2008	No	No
Derivation control	x	x	-	x	x	x
Multiple objects	-	x	x	x	-	-
Object types	Lsystems	dxf, Lsystems	Plants Fractals	3D Objects	Trees	Trees
Textures	-	x	-	x	x	x
Purpose	Fractal trees	Lsystems	Plants	Ivy Plants	Trees	Trees
Usability	6	8	8	9	8	7
Documentation	6	2	10	4	6	0
Debugging	Level control	-	-	Level control	Instant update	Level control
Modeling Speed	1	1	2	3	4	5

Table 1. (*continued*)

	MeshTree Studio	Dryad	Arbaro	XFrog	SpeedTree	Tree[d]
2D	-	-	-	-	-	-
3D	x	x	x	x	x	x
IDE	x	x	x	x	x	x
Definition Language	GUI	GUI	GUI	GUI	GUI	GUI
Navigation	x	x	x	x	x	x
Geometry Generation	x	x	x	x	x	x
Image Generation	-	-	x	x	x	x
Import	-	-	xml	-		-
Export	.mesh	obj	obj, povray, dxf	png, jpg	several	x, b3d
License	Freeware	Freeware	Freeware	Trial, lite, full	Trial, full	Freeware
Price	0	0	0	$300/$400	$8495,	0
Released	2007	2007	2003	1996	2002	2002
Updated	No	2008	2004	2002	2009	2008
Derivation control	x	-	x	x	x	-
Multiple objects	-	-	-	x	x	-
Object types	Trees	Trees	xml	Plants	Plants	Trees
Textures	x	-	-	x	x	x
Purpose	Trees	Trees	Trees	Plants	Plants	Trees
Usability	6	5	5	9		5
Documentation	0	0	6	10		0
Debugging	Level control	Instant update	Instant update	a lot		Instant update
Modeling Speed	7	5	6	2	8	5

References

1. Parish, Y.I.H., Muller, P.: Procedural modeling of cities. In: SIGGRAPH 2001, pp. 301–308 (2001)
2. Greuter, S., Parker, J., Stewart, N., Leach, G.: Real-time procedural generation of 'pseudo infinite' cities. In: GRAPHITE 2003, pp. 87–94 (2003)
3. Sun, J., Yu, X., Baciu, G., Green, M.: Template-based generation of road networks for virtual city modeling. In: ACM Symposium on Virtual Reality Software and Technology, pp. 33–40 (2002)
4. Muller, P., Wonka, P., Haegler, S., Ulmer, A., Gool, L.V.: Procedural modeling of buildings. In: SIGGRAPH 2006, pp. 614–623 (2006)
5. Wonka, P., Wimmer, M., Sillion, F., Ribarsky, W.: Instant architecture. ACM Transactions on Graphics 22(3), 669–677 (2003)
6. Martin, J.: Procedural house generation: A method for dynamically generating floor plans. In: Symposium on interactive 3D Graphics and Games (2006)
7. Ebert, D., Musgrave, F., Peachey, D., Perlin, K., Worley, S.: Texturing and Modeling: A Procedural Approach, 3rd edn. Morgan Kaufmann, San Francisco (2002)
8. Weber, J., Penn, J.: Creation and rendering of realistic trees. In: SIGGRAPH 1995, pp. 119–128. ACM, New York (1995)
9. Lluch, J., Camahort, E., Vivó, R.: Procedural multiresolution for plant and tree rendering. In: AFRIGRAPH 2003 (2003)
10. Roden, T., Parberry, I.: Clouds and stars: efficient real-time procedural sky rendering using 3d hardware. In: ACE 2005 Int. Conference on Advances in Computer Entertainment Technology, pp. 434–437 (2005)
11. Prachyabrued, M., Roden, T.E., Benton, R.G.: Procedural generation of stylized 2d maps. In: ACE 2007: Advances in Computer Entertainment Technology (2007)
12. Roden, T., Parberry, I.: Procedural Level Generation. In: Game Programming Gems 5, pp. 579–588. Charles River Media (2005)
13. Quan, L., Tan, P., Zeng, G., Yuan, L., Wang, J., Kang, S.B.: Image-based plant modeling. In: SIGGRAPH 2006, pp. 599–604 (2006)
14. Deix, W.: Real-time rendering of fractal rocks. In: Central European Seminar on Computer Graphics (2003)
15. Prusinkiewicz, P., Hammel, M.: A fractal model of mountains with rivers. In: Graphics Interface 1993 (1993)
16. Lindenmayer, A.: Mathematical models for cellular interaction in development, parts i and ii. Journal of Theoretical Biology 18, 280–315 (1968)
17. Kelly, G., McCabe, H.: A survey of procedural techniques for city generation. ITB Journal 14 (2006)
18. Prusinkiewicz, P., Hammel, M.S., Mjolsness, E.: Animation of plant development. In: SIGGRAPH 1993, pp. 351–360 (1993)
19. FracTree: `http://archives.math.utk.edu/software/msdos/fractals/fractree`
20. L-System4: `http://www.geocities.com/tperz/L4Home.htm`
21. LStudio: `http://algorithmicbotany.org`
22. An Ivy Generator: `http://graphics.uni-konstanz.de/~luft/ivy_generator`
23. TreeGenerator: `http://www.treegenerator.com`
24. TreeMagik G3: `http://www.aliencodec.com/product_treemagik.php`
25. MeshTree Studio: `http://www.ogre3d.org/forums/viewtopic.php?t=25909`
26. Dryad: `http://dryad.stanford.edu`
27. Arbaro: `http://arbaro.sourceforge.net`
28. SpeedTree: `http://www.speedtree.com`
29. Xfrog: `http://www.xfrog.com/`
30. Tree[d]: `http://www.frecle.net/forum/viewtopic.php?t=780`

Toward the New Generation of Intelligent Distributed Computing Systems

Robert Schaefer[1], Krzysztof Cetnarowicz[1], Bojin Zheng[2], and Bartłomiej Śnieżyński[1]

[1] Department of Computer Science
AGH University of Science and Technology, Krakow, Poland
[2] College of Computer Science
South-Central University for Nationalities, Wuhan, China

Abstract. This paper is an introduction to the works presented in Intelligent Agents and Evolvable Systems Workshop. The workshop focuses on the various applications of agent-oriented systems, the roles of evolution and interactions of agents to build intelligent systems.

Agent-oriented system is the new attractive tool for high performance distributed processing. Agent-oriented programming comprehends the ability to integrate the resources of computer networks with the flexibility of their governing. Software agent technology constitutes also the powerful tool for solving various decentralized decision making and technological problems. Multi-agent systems have many connections to evolutionary computation. Evolutionary system can be implemented in an agent-oriented fashion. Evolution is also regarded as one of fundamental forms of adaptation of intelligent agents. The workshop *Intelligent Agents and Evolvable Systems* focuses on the various applications of agent-oriented systems, the roles of evolution and interactions of agents to build intelligent systems.

The first group of papers presented in the workshop concerns theoretical aspects of multi-agent systems, their formal descriptions, architectures, and applications of such systems to solving various engineering tasks as well.

Cetnarowicz in "From Algorithm to Agent" tries to introduce the relation between the notion of the algorithm and the concept of the agent. His approach helps to develop more formal description of the agent, and crucial properties of the agent and the cooperative multi-agent system.

The paper "Agent-Based Model and Computing Environment Facilitating the Development of Distributed Computational Intelligence Systems" by Byrski and Kisiel-Dorohinicki proposes a simple formalism that describes the hierarchy of multi-agent systems, which is particularly suitable for the design of a certain class of distributed computational intelligence systems. The mapping between the formalism and the existing computing environment AgE is also sketched out.

Śnieżyński in "Agent Strategy Generation by Rule Induction in Predator-Prey Domain" shows the new rule induction method for generating artificial agent strategy. This method was tested on a predator-prey domain. Experimental results show that the learning process is fast.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 813–814, 2009.

The next contribution "Multi-agent system for recognition of hand postures" by Jurek, Flasiński and Myśliński show how we can apply the multi-agent system to develop a system for recognition of hand postures of the Polish Sign Language using a syntactic pattern recognition approach. They propose to make a construction of a grammar within a parsable class ETPL(k) dubious.

Koźlak, Konieczny, and Żabińska in "Multi-agent crisis management in transport domain" present a multi-agent system for crisis management in transport domain. The system enables to solve static and dynamic versions of transport problem such as vehicle failures and traffic jams.

The paper "Agent-based environment for knowledge integration" written by Koźlak, Zygmunt and Siwik shows an environment for the integration of knowledge expressed with the use of the ontology description languages. Presented approach may enable to obtain access to services, which offer knowledge contained in various distributed databases associated with semantically described web portals.

The last paper of this group "The norm game – how a norm fails" by Kułakowski, Dydejczyk and Rybak presents the simulations of the norm game between players at nodes of a directed random network. The authors suggested that the final boldness, i.e. the probability of norm breaking by the players, can vary with the threshold value of the initial boldness which can be interpreted as a norm strength. The simulation results are discussed in the context of the statistical data on crimes, divorces and on the alcohol consumption.

Another group of papers is devoted to algorithms solving hard engineering problems controlled by the linguistic or biological and social inspired mechanisms.

Paszyński, Paszyńska and Grabska in "Graph transformations for modeling hp-adaptive Finite Element Method with mixed triangular and rectangular elements" present a new approach that allows to control the iterative, adaptive process of solving partial differential equations by the system of graph grammar productions. Such a linguistic model helps to exploit the concurrency in the each step of the iteration. It may be also used for profiling the computation in the cluster environment in order to gain the maximum speedup.

In the next paper of this group "Graph grammar based Petri nets model of concurrency for self-adaptive hp-Finite Element Method with triangular elements" by Paszyński, Paszyńska and Szymczak the Petri net is used as a model of the graph grammar production scheduling. The similar technique of hp–FEM computation controlling as in the previous paper is applied in the case of rectangular meshes.

Barabasz, Schaefer and Paszyński in "Handling ambiguous inverse problems by the adaptive genetic strategy hp–HGS" deliver the socially inspired, multideme, genetic algorithm for effective solving of inverse, parametric problems for partial differential equations. The accuracy of solving direct problem is dynamically adjusted to the accuracy of evaluating inverse problem error, which additionally decreases the total computational cost.

Multi-agent System for Recognition of Hand Postures

Mariusz Flasiński, Janusz Jurek, and Szymon Myśliński

IT Systems Department, Jagiellonian University
Straszewskiego 27, 31-110 Cracow, Poland
http://www.wzks.uj.edu.pl/ksi

Abstract. A multi-agent system for a recognition of hand postures of the Polish Sign Language is presented in the paper. The system is based on a syntactic pattern recognition approach, namely on parsable ETPL(k) graph grammars. An occurrence of a variety of styles of performing hand postures requires an introduction of many grammar productions that differ each from other slightly. This makes a construction of a grammar within a parsable class ETPL(k) dubious. Dividing a whole grammar into sub-grammars and distributing them to agents allows one to solve the problem.

Keywords: pattern recognition, multi-agent system, ETPL(k) graph grammar.

1 Introduction

Multi-agent systems are used as a useful approach in a pattern recognition area [11,13,21]. It concerns, especially, real-time applications. However, real-time requirements make a construction of multi-agent systems difficult [15,20,22]. The results of our recent research has revealed that a *syntactic* pattern recognition paradigm together with a multi-agent system approach gives a convenient basis for solving real-time pattern recognition problems [9,16,17]. At the same time, it seems that in an aspect of an efficiency of a symbolic description processing it performs better than many other computationally inefficient standard artificial intelligence methods [19].

The main idea of syntactic pattern recognition consists in treating a pattern as a structure of the form of string, tree or graph. A set of patterns to be recognized is treated, in turn, as a formal language (string language, tree language or graph language) that is generated with a formal grammar. A formal automaton (syntax analyzer, parser) constructed for the grammar is then used as a recognition algorithm. The *ETPL*(k) graph grammars (*Embedding Transformation-preserving Production-ordered k-Left nodes unambiguous* grammars) [2,4,7,10] have been used as a syntactic pattern recognition model for a variety of applications such, as: a pattern recognition in the industrial robot control system [4], an analysis of a distributed environment configuration in the software allocation system [3], a solid modelling and analysis in the CAD/CAM (*Computer Aided*

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 815–824, 2009.

Design / Computer Aided Manufacturing) integration system [5], a solid representation for an optimal scaling in the CAE (*Computer Aided Engineering*) parallel computation system [6], an analysis of a complex experimental physics equipment in the real-time expert control system [8,16]. Analogously like in the last mentioned area (real-time expert control system), we have applied a multi-agent approach during our recent research concerning a real-time recognition of hand postures occurring in the Polish Sign Language. The results of the research are presented in the paper.

Let us notice that pattern recognition of hand postures is widely investigated all over the world [12,14,18]. The methods of hand posture recognition can be applied in the field of gesture recognition [1,18]. A distributed recognition approach is also widely used for recognition tasks to boost system performance and/or recognition rate [1,23,24].

A model of a syntactic pattern recognition-based agent is introduced in the next chapter. An architecture of the multi-agent system and scenarios of its functioning are discussed in chapter 4. The final chapter contains concluding remarks.

2 Syntactic Pattern Recognition-Based Agents

We introduce a model of a syntactic pattern recognition-based agent in this chapter. In section 2.1 we present basic phases of a syntactic pattern recognition approach to an analysis of hand postures and a formal model based on ETPL(k) graph grammars [2,4,7,10]. Then, in the same section, we discuss a basic problem concerning a construction of such a grammar for the Polish Sign Language. Introducing a multi-agent architecture of a system allows us just to solve the problem. An internal architecture of a syntactic pattern recognition-based agent that is defined according to a formal model introduced is presented in section 2.2.

2.1 A Formal Model for a Syntactic Pattern Recognition-Based Agent

An analysis of hand postures of the Polish Sign Language consists of three main phases. During the first phase of image preprocessing a hand region is identified and its contour is approximated with a polygon as it is shown in Figures: 1(a) and 1(b). The region centroid and the polygon vertices constitute the characteristic points of the image - see Fig. 1(b). The image characteristic points are then used, at the second phase of a graph description generation, for spanning a graph structure, which is depicted in Fig. 1(c), and finally for defining the so-called IE-graph on the basis of this structure - see Fig. 1(d). Let us introduce a formal definition of an IE-graph [4,7].

Definition 1. An *indexed edge-unambiguous graph*, *IE* graph, over Σ and Γ is a quintuple

$$g = (V, E, \Sigma, \Gamma, \phi),$$

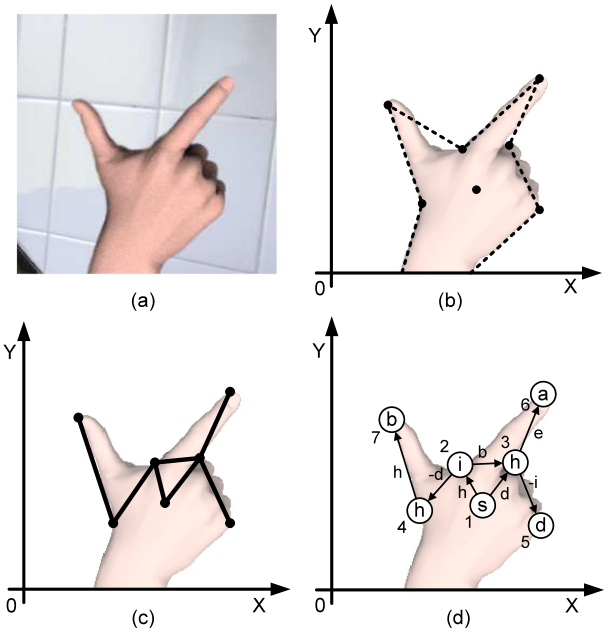


Fig. 1. Main phases of a syntactic pattern recognition of hand postures of the Polish Sign Language

where V is a finite, non-empty set of nodes that indices have been ascribed to in an unambiguous way,
Σ is a finite, non-empty set of node labels,
Γ is a finite, non-empty set of edge labels,
E is a set of edges of the form (v, λ, w), where $v, w \in V, \lambda \in \Gamma$, such that index of v is less than index of w,
$\phi : V \longrightarrow \Sigma$ is a node-labelling function,
and g contains a BFS spanning tree, which nodes are indexed in the same way as nodes of g and its edges are directed in the same way as edges of g.

IE-graphs belong to a parsable class of ETPL(k) graph grammars [2,4,7]. The ETPL(k) parsing algorithm is of a very good computational complexity, namely $O(n^2)$ [4]. The parsing algorithm uses an ETPL(k) graph grammar (constructed for a given graph language consisting of IE-graphs representing hand postures) as its control table. Of course, constructing such a grammar "by hand" is very difficult in case a language consists of many graphs. Therefore, an inference algorithm generating, automatically, such a grammar on a basis of a set of IE-graphs has been defined and implemented as the INFERGRAPH system [10]. Such a method has occurred to be very efficient.

In case of a recognition of hand postures performed carefully by signers that use a standard (model) Polish Sign Language, a very good recognition rate has been obtained - ca. 95%. However, in case of signers representing a variety of

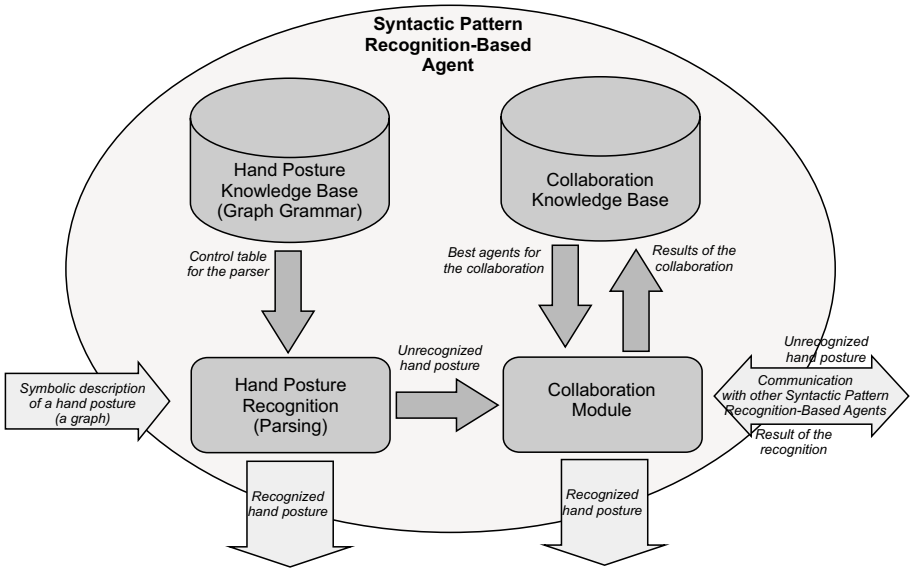


Fig. 2. A general scheme of a syntactic pattern recognition-based agent

(incorrect, negligent) styles (manners), a grammar should contain a lot of productions that differ each from other slightly. This phenomenon makes a definition of a grammar within a parsable class ETPL(k) dubious. In case of an automatic generation of a grammar with the inference system, it can cause a need of introducing a large number of productions that decreases a time efficiency of the method. In extreme cases, it can cause a generation of a grammar within a parsable class ETPL(k) even impossible (if we would like to comprise all the distorted performances of the sign language).

Dividing the whole grammar into sub-grammars corresponding to various styles and manners of a sign language performance allows us to solve a problem discussed above. Then, each such a "specific" sub-grammar is allocated to a syntactic pattern recognition-based agent that is responsible for recognizing a set of hand postures performed in one predefined manner. An architecture of such an agent is discussed in the next section.

2.2 An Architecture of a Syntactic Pattern Recognition-Based Agent

There are following main elements of a syntactic pattern recognition-based agent (see Fig. 2):

- a *hand posture knowledge base* in the form of the ETPL(k) graph grammar describing a manner (variant) of performing hand postures,
- a *parser* analyses IE-graph representations of hand postures on a basis of the ETPL(k) graph grammar (a hand posture knowledge base),

- a *collaboration module* is responsible for: collaborating with other syntactic pattern recognition-based agents (establishing communication, exchanging data, etc.) and accumulating knowledge concerning results of such a collaboration,
- a *collaboration knowledge base* stores knowledge concerning results of a collaboration with other syntactic pattern recognition-based agents (constantly being updated). Let us notice that the collaboration scenarios are described in details in chapter 3.

An input for an agent is of the form of an IE-graph being a symbolic representation of a hand posture. A recognition of a hand posture is made with parsing of a received IE-graph (on the basis of an ETPL(k) graph grammar associated with the agent). The result is outputted to the control agent (see: chapter 3).

When an input IE-graph cannot be recognized by the parser, then the agent starts a collaboration with other syntactic pattern recognition-based agents (via *collaboration module*). The agent sequentially passes the graph to other agents asking for its recognition until it gets a positive answer, i.e. a result of the recognition.

Syntactic pattern recognition-based agents are provided with a self-learning mechanism. Learning is made by a continuous accumulation of knowledge concerning results of the collaboration with other syntactic pattern recognition-based agents.

Each agent has its own *collaboration knowledge base*. It is built in the form of the following table.

Table 1. Collaboration knowledge base of an agent

Agent id.	Technical (communication) information	Number of queries	Number of positive answers	Percentage of positive answers
*A*_01	*active, port: xxx*	2	1	50%
*A*_02	*inactive, port: yyy*	3	0	0%
*A*_03	*active, port: zzz*	10	8	80%
...	...	...	...	...

Information about collaboration with a particular agent is stored in the line of the table. The first column of the table contains technical information needed to communicate with a given agent. The second one contains the number of all queries send to a given agent. The third column contains the number of queries which returned a positive answer (a hand posture has been recognized). The last column contains the percentage of positive answers in case of a given agent.

The table is updated each time the agent asks for the help of another syntactic pattern recognition-based agent. The knowledge included in the table is a basis for a collaboration strategy of the particular agent. A collaboration strategy is described in the next chapter.

At the end of this section, we would like to underline the autonomy and pro-social behavior of the syntactic pattern-recognition based agents. Therefore, let us discuss how the BDI standards (Belief-Desire-Intention) are embedded in the agents.

- *Belief.* The belief of an agent is represented by the collaboration knowledge base. Let us notice that the base can be changed. It reflects the fact that what an agent believes may not necessarily be true (it may change in the future).
- *Desires.* The only desire (goal) of an agent is to recognize a hand posture (being the input data). An agent tries to accomplish the goal by itself or in collaboration with other agents.
- *Intentions.* The intention of an agent is to recognize a hand posture as quick as possible. If its own parsing algorithm fails, then it decides which other agent is "the best" to cooperate with. The decision is based on the information stored in the collaboration knowledge base (see: section 3.1).

3 Architecture of the Multi-agent System and Scenarios of its Functioning

An architecture of the multi-agent system recognizing hand postures is presented in Fig. 3.

Let us assume that there are n syntactic pattern recognition-based agents, where each agent is responsible for recognizing hand postures performed in a specific manner. The multi-agent system consists of: several *active* syntactic pattern

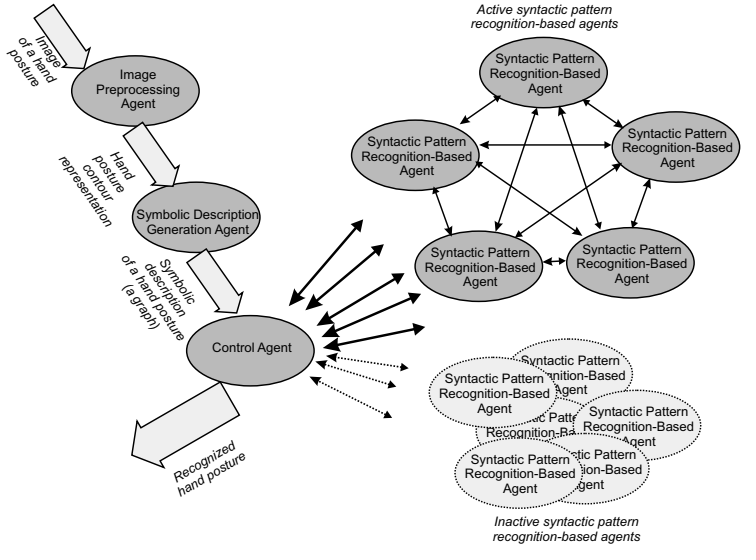


Fig. 3. Architecture of the multi-agent system

recognition-based agents, the remaining *inactive* syntactic pattern recognition-based agents, the control agent, the image preprocessing agent and the symbolic description generation agent (the last two agents are responsible for preliminary phases of image processing that have been discussed in section 2.1).

Active agents are running at processors available for the system. All active agents can communicate with each other. *Inactive* agents are not allocated to the processors. They wait to be activated by the control agent.

The control agent is responsible for the following tasks:

- a communication with the "outside world" (i.e. collecting hand postures to be recognized, and outputting the results of a recognition),
- an allocation of syntactic pattern recognition-based agents to the processors,
- a communication with syntactic pattern recognition-based agents (a distribution of hand posture IE-graph representations among agents, and gathering results of a recognition),
- an evaluation of a performance of syntactic pattern recognition-based agents and a re-allocation of agents (as a consequence of the evaluation).

In order to perform all tasks mentioned above, the control agent has its own *collaboration knowledge base*, which is constructed in the same way like the one of a syntactic pattern recognition-based agent (see: Table 1). However, a scenario of supplying the base with new information is different from that of syntactic pattern recognition-based agents.

Now, let us present functioning of the system. Initially, the control agent starts as much syntactic pattern recognition-based agents as it is possible in the system. Let m is a number of such agents. Then, the control agent initializes its own *collaboration knowledge base* and the bases of all active agents (providing information on technical ways of a communication — see: column 2 in Table 1).

In the next step, the control agent begins to receive data from the "outside world". The data are consecutive IE-graph representations of hand postures to be recognized. The agent distributes m hand postures among active syntactic pattern recognition-based agents and waits for their response. The remaining received hand postures are buffered in the FIFO (first-in first-out) structure.

In the next two sections we discuss a further functioning of syntactic pattern recognition-based agents as well as a way of evaluating a performance of these agents and re-allocating these agents made by the control agent.

3.1 Functioning of Syntactic Pattern Recognition-Based Agents

There are two following scenarios of functioning of a syntactic pattern recognition-based agent.

Scenario 1. A request to recognize a hand posture is made by the control agent. Firstly, an agent tries to recognize a hand posture itself. An analysis of the hand posture is made by a parser of the agent. If a result of the analysis is positive (the hand posture is recognized), the agent returns information to the control agent which passes it to the "outside world". If the result is negative (the

hand posture cannot be recognized), the agent starts a collaboration with other syntactic pattern recognition-based agents.

A strategy of a collaboration is straightforward. The agent sorts its collaboration knowledge base accordingly to a percentage of positive answers given by other agents (see: column 4 in Table 1). A request to recognize a hand posture is firstly sent to the "best rated" *active* agent. If an answer is negative, the request is directed to the next active agent in ranking, and so on. A procedure is repeated until the agent gets a positive answer or all active agents have returned negative answers. During this process the agent collects all information about the collaboration in its knowledge base. Finally, information concerning the responses of the collaborating agents is passed to the control agent.

If a positive answer is finally achieved, it ends the scenario. Otherwise, the agent asks the control agent to activate the remaining agents (which are now in an inactive state) and a recognition process starts again.

Scenario 2. A request to recognize a hand posture is made by another syntactic pattern recognition-based agent.
The agent tries to recognize a hand posture itself, and returns an answer (positive or negative) to an agent that has made the request. In this scenario the agent does not start any collaboration with other agents.

3.2 Functioning of the Control Agent

There are two following scenarios of functioning of the control agent.

Scenario 1. A re-allocation as a result of an evaluation of syntactic pattern recognition-based agents.
This is a typical scenario. After receiving an answer from a syntactic pattern recognition-based agent, the control agent updates its collaboration knowledge base, and checks percentages of positive answers for each active agent (in this way the performance of each active agent is evaluated). If there is an active agent A which rank is below a rank of the best inactive agent N, then A is replaced with N. It means that now A agent becomes inactive, and N agent becomes active and it starts running at a given processor.

Scenario 2. A re-allocation as a result of the lack of a recognition.
The second scenario takes place, if a hand posture cannot be recognized by any active agent in the system. In such an unusual situation, the control agent replaces all active agents with the "best" inactive agents.

4 Concluding Remarks

A multi-agent system for an identification of hand postures of the Polish Sign Language has been implemented on the basis of a syntactic pattern recognition approach discussed in the paper. Experimental testing (more than 700 tests) has

revealed its good characteristics. The average recognition of a hand posture takes 0.13 s (C++, *Intel Core Duo, 1.83 GHz, 1GB RAM*), and a recognition rate is of 95%. Including examples of a hand posture performance negligence resulting from specific (incorrect) styles of signing to a sample set makes a graph grammar generating this set very complex and difficult to be maintained.

The main advantage of the use of a multi-agent system approach presented in the paper consists in a possibility of dividing the graph grammar generating IE-graph representations of hand postures into several sub-grammars. It, in turns, allows us to handle a problem of a complexity of a control table of the ETPL(k) parser [4]. Such a distributed control table is, obviously, much easier to be maintained with the grammatical inference method [10].

The multi-agent approach not only influences the possibility of constructing a suitable graph grammar (the quality aspect of the system), but also the efficiency of the system. Experiments performed on both single-agent system and 4-agent system show ca. 28% increase in time efficiency in favor of the multi-agent approach.

There are self-learning mechanisms embedded in a system defined in such a way. Firstly, the control agent learns, which syntactic pattern recognition-based agents provide the best results in case of a particular signer performing hand postures (such agents should use our computing resources firstly, in case of a given signer). Secondly, each syntactic pattern recognition-based agent learns which agents are the best in the collaboration. (It means that the agents can learn the best strategies of a recognition of hand postures). As we have already noticed, styles of showing postures can be combined. It means that an agent that is the best one in a given collaboration is not necessarily the best agent in the global ranking.

A syntactic pattern recognition method presented in the paper is the deterministic one. In case of hand posture images that are distorted (fuzzy), a probabilistic or fuzzy graph grammars would give better recognition rates. It would require to represent also a collaboration knowledge used by the system in a probabilistics (or fuzzy) way. The results of a research into such an extension of the method will be a subject of further publications.

References

1. Bauer, B., Kraiss, K.F.: Towards an automatic sign language recognition system using subunits. In: Wachsmuth, I., Sowa, T. (eds.) GW 2001. LNCS, vol. 2298, pp. 64–75. Springer, Heidelberg (2002)
2. Flasiński, M.: Parsing of edNLC-graph grammars for scene analysis. Pattern Recognition 21, 623–629 (1988)
3. Flasiński, M., Kotulski, L.: On the use of graph grammars for the control of a distributed software allocation. The Computer Journal 35, A165–A175 (1992)
4. Flasiński, M.: On the parsing of deterministic graph languages for syntactic pattern recognition. Pattern Recognition 26, 1–16 (1993)
5. Flasiński, M.: Use of graph grammars for the description of mechanical parts. Computer Aided Design 27, 403–433 (1995)

6. Flasiński, M., Schaefer, R., Toporkiewicz, W.: Optimal stochastic scaling of CAE parallel computations. In: Polkowski, L., Skowron, A. (eds.) RSCTC 1998. LNCS, vol. 1424, pp. 557–564. Springer, Heidelberg (1998)
7. Flasiński, M.: Power properties of NLC graph grammars with a polynomial membership problem. Theoretical Computer Science 201, 189–231 (1998)
8. Flasiński, M., Jurek, J.: Dynamically programmed automata for quasi context sensitive languages as a tool for inference support in pattern recognition-based real-time control expert systems. Pattern Recognition 32, 671–690 (1999)
9. Flasiński, M.: Automata-based multi-agent model as a tool for constructing real-time intelligent control systems. In: Dunin-Keplicz, B., Nawarecki, E. (eds.) CEEMAS 2001. LNCS, vol. 2296, pp. 103–110. Springer, Heidelberg (2002)
10. Flasiński, M.: Inference of parsable graph grammars for syntactic pattern recognition. Fundamenta Informaticae 80, 379–413 (2007)
11. Hakeem, A., Shah, M.: Learning, detection and representation of multi-agent events in videos. Artificial Intelligence 171, 586–605 (2007)
12. Holden, E.J., Owens, R.: Visual sign language recognition. In: Klette, R., Huang, T.S., Gimel'farb, G. (eds.) Dagstuhl Seminar 2000. LNCS, vol. 2032, pp. 270–287. Springer, Heidelberg (2001)
13. Hongeng, S., Nevatia, R.: Multi-agent event recognition. In: Proceedings of the Eight International Conference on Computer Vision, Vancouver, Canada, July 07–14, vol. 2, pp. 84–91 (2001)
14. Huang, T.S., Pavlovic, V.: Hand gesture modeling, analysis and synthesis. In: Int. Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland (1995)
15. Julian, V., Carrascosa, C., Rebollo, M., Soler, J., Botti, V.: SIMBA: An approach for Real-Time Multi-agent Systems. In: Escrig, M.T., Toledo, F.J., Golobardes, E. (eds.) CCIA 2002. LNCS (LNAI), vol. 2504, pp. 282–293. Springer, Heidelberg (2002)
16. Jurek, J.: Syntactic pattern recognition-based agents for real-time expert systems. In: Dunin-Keplicz, B., Nawarecki, E. (eds.) CEEMAS 2001. LNCS, vol. 2296, pp. 161–168. Springer, Heidelberg (2002)
17. Jurek, J.: Grammatical inference as a tool for constructing self-learning syntactic pattern recognition-based agents. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 712–721. Springer, Heidelberg (2008)
18. Krüger, M., von der Malsburg, C., Würtz, R.P.: Self-organized evaluation of dynamic hand gestures for sign language recognition. In: Würtz, R.P. (ed.) Organic Computing. Springer, Heidelberg (2008)
19. Negnevitsky, M.: Artificial Intelligence. A Guide to Intelligent Systems. Addison-Wesley, Reading (2002)
20. Niederberger, C., Gross, M.: Hierarchical and heterogenous reactive agents for real-time applications. Computer Graphics Forum 22, 323–331 (2003)
21. Rodin, V., Benzinou, A., Guillaud, A., Ballet, P., Harrouet, F., Tisseau, J., Le Bihan, J.: An immune oriented multi-agent system for biological image processing. Pattern Recognition 37, 631–645 (2004)
22. Soto, I., Garijo, M., Iglesias, C.A., Ramos, M.: An agent architecture to fulfill real-time requirements. In: Proceedings of the Fourth International Conference on Autonomous Agents, Barcelona, Spain, June 03–07, pp. 475–482 (2000)
23. Würtz, R.P. (ed.): Organic Computing. Springer, Heidelberg (2007)
24. Yang, X., Xing, Y., Nguyen, A.: A Web-Based Face Recognition System Using Mobile Agent Technology. In: The 8th Australian World Wide Web Conference (2002)

From Algorithm to Agent

Krzysztof Cetnarowicz

AGH University of Science and Technology, Krakow, Poland
cetnar@agh.edu.pl

Abstract. Although the notion of an agent has been used in computer science for a dozens of years now it is still not very well defined. It seems that there is a lack of formal definition of such concepts as an "object" and an "agent". It makes difficult formal analysis of algorithms developed with their use. We should find more formal description that has connection with the basic definition of the algorithm.

In the paper we will propose an approach that may help to develop more formal definitions of an agent and an object with the use of algorithm concept. Starting from the notion of the algorithm and using the observation that complex algorithm should be developed with the use of its decomposition we propose some ideas how we can consider such notions as object and agent.

Proposed approach takes into consideration the necessity of the autonomy and of an agent and object and the problems of the interactions between them and suggest the resolution of the problems by communication and observation process.

Presented concept of an object and an agent makes possible to find further more formal definitions of these notions and find the crucial properties of these concepts and the main difference between the notion of an object and the notion of an agent.

1 Introduction

Development of programming techniques and methods of software construction may be viewed as the development of algorithms working in a given environment ([14]). The process of creation of new more powerful algorithms resulted in definition of new elements that may be used to build them. Algorithms became more complex and more difficult to be developed. So, according to the principle: "divide and conquer" we will decompose the algorithm and the development process.

Although the notion of an agent has been used in computer science for dozens of years now it is still not very well defined. It seems that there is a lack of formal definition of such concepts as an "object" and an "agent". It makes formal analysis of algorithms developed with the use of concept of an object or an agent difficult ([10], [11], [12]).We should find better formal description that has connection with the base definition of the algorithm.

In the paper we will propose an approach that may help to develop more formal definitions of an agent and an object within the concept of an algorithm.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 825–834, 2009.

Starting from the notion of an algorithm and using the observation that complex algorithm should be developed with the use of its decomposition we propose some ideas how we can consider such notions as object and agent. Proposed approach takes into consideration the necessity of the autonomy of an agent and an object and the problems of the interactions between them and suggest the resolution of the problems by communication and observation process. ([1], [2], [5], [7]).

Considering non-formal, based on intuition point of view of the concept of an algorithm we can say that an algorithm describes activity of an object or an agent in a given environment ([4], [8], [14]). These were programmers who noticed that current algorithms are hard to be developed because of their complexity, and the way to simplify the creation process is to use the previously mentioned method "divide and conquer". A number of approaches to divide (or rather decompose) an algorithm had been invented and applied. The idea of an object and then of an agent may be considered as the result of an attempt to enable the decomposition of an algorithm. Presented here approach to the notion of an object and an agent makes possible to find properties of these concepts and the difference between them.

2 Algorithm and Decomposition

We can consider the following definition of the algorithm ([13]):

$$Alg = (U, F) \tag{1}$$

where:

$$U - set, \quad U \neq \emptyset, \quad F : \ U \rightarrow U \tag{2}$$

Function F is a partial function. It means that the domain of the function F is a subset of the set U. Elements u of the set U are called states of the algorithm Alg.

Realization (execution) of the algorithm Alg for a given initial state u^0 may be considered as a finite sequence [3]:

$$u^0, \ u^1, \ ... \ u^i, \ u^{i+1}, \ ... \ u^k \tag{3}$$

or infinite:

$$u^0, \ u^1, \ ... \ u^i, \ u^{i+1}, \ ... \tag{4}$$

such, that

$$u^{i+1} = F(u^i) \tag{5}$$

The sequence is finite when there is a final state u^k. There is a final state when u^k belongs to the $Im(F)$ and does not belong to the $Dom(f)$. So, the final states of the algorithm Alg are elements of the set U that do not belong to the domain of the function F.

To determine elements u of the set U is usually realized by characteristic properties of the elements u. We can consider elements of the set U as n-tuples of characteristic parameters [6]:

$$u^k = (x_1^k, x_2^k, ..., x_m^k) \tag{6}$$

where every element x_j^i determines a characteristic property j of the element u^i. Instead of the set U we use the set X of n-tuples $(x_1^k, x_2^k, ..., x_m^k)$.

When we consider the function F:

$$F(u^k) = (u_1^{k+1}) \tag{7}$$

and using the following expressions:

$$u^k = (x_1^k, x_2^k, ..., x_m^k), \quad u^{k+1} = (x_1^{k+1}, x_2^{k+1}, ..., x_m^{k+1}) \tag{8}$$

we can find that the function F may be replaced by the partial function $f : X \to X$, which operates on the mentioned above characteristic properties:

$$f(x_1^k, x_2^k, ..., x_m^k) = (x_1^{k+1}, x_2^{k+1}, ..., x_m^{k+1}) \tag{9}$$

In practical applications there is a great number of characteristic properties that define the states of the algorithm and the function f is complex. Consequently, it is difficult to study and develop the algorithm. In such a case the best way is to decompose the the algorithm.

The decomposition of the algorithm Alg concerns the set of characteristic properties $(x_1^i, x_2^i, ..., x_m^i)$, and the function f. As the result of such decomposition we have to obtain the decomposition of the complex algorithm Alg into a number of partial algorithms $Alg_1, Alg_2, ...Alg_n$.

The main goal of the decomposition is to enable the separate development of every partial algorithm. Then the complex algorithm is built as composition of the partial algorithms. Let us consider the algorithm $Alg = (X, f)$ defined by the set of states (or characteristic properties), and the function f realizing the evolution of the algorithm. So we have:

$$Alg = (X, f), \quad f : \ X \to X. \tag{10}$$

Let the algorithm Alg is decomposed into partial algorithms what means that instead the algorithm Alg we consider a number of partial algorithms. To simplify our analysis we can consider that we have two partial algorithms Alg_r and Alg_s. Using the partial algorithms Alg_r and Alg_s we may obtain the same results as the with the use of the algorithm Alg.

The decomposition may concern the function f and/or the set X. Let us consider as a first approach only the decomposition of the function f. The function f may be decomposed into two partial functions f_r and f_s by the following way:

$$f = f_r \cup f_s, \quad f_r \cap f_s = \emptyset \tag{11}$$

$$f_r : \ X \to X, \quad Alg_r = (X, f_r) \tag{12}$$

$$f_s : \ X \to X, \quad Alg_s = (X, f_s) \tag{13}$$

The successive application of functions f_r and f_s enables to transform the starting state of the algorithm to the final one that represents a needed solution. But the decomposition of only the function f is not sufficient to enable the separate

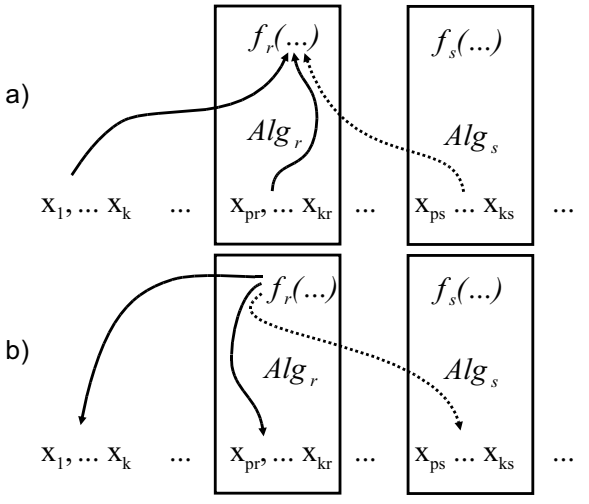


Fig. 1. Schema of the decomposition of characteristic properties

development of partial algorithms. We have to decompose the set X of the algorithm Alg. Let us consider following sets of properties of the algorithms: X_r - representing properties of the algorithm Alg_r, X_s - properties of the algorithm Alg_s and the set X_0 describing resources of the environment that are accessible for every partial algorithm. The set X is decomposed by the following way:

$$X = X_0 \times X_r \times X_s \tag{14}$$

The set X_r (X_s - respectively) represents properties of the algorithm Alg_r (Alg_s - respectively) and for the sake of separate development of the algorithms Alg_r and Alg_s we have to enclose hermetically the properties of every algorithm. As a result of that the algorithm Alg_r has no more access to the parameters X_s and vice versa (dash arrow on the fig. 1).

We can consider two solutions proposed for this problem. One basing on the communication paradigm that leads us to the object concept, and the next one using operation of the observation that gives us the concept on agent.

3 Decomposition of an Algorithm Using the Concept of an "Object"

We can consider the decomposition of the set of characteristic properties. Every partial algorithm has its own subset of characteristic properties. A given partial algorithm r (Alg_r) is defined by the function f_r and the set of characteristic properties: $\{x_{p_r}, x_{p_r+1}, ..., x_{k_r}\}$ that is a subset of the global set of characteristic properties: $\{x_1, x_2, ..., x_m\}$. The characteristic properties of one partial algorithm

(for instance Alg_r) may be used by the function of this algorithm (f_r)and are not accessible to another algorithm (for instance Alg_s) what means that they may not be used by the function f_s (fig. 1). Such consideration brought us to the concept of an "object" [13].

We can say that the partial algorithm Alg_r may be considered as an object Obj_r, the state of the object is described by parameters $\{x_{p_r}, x_{p_r+1}, ..., x_{k_r}\}$ and the evolution of the state of this object is realized by the function f_r^o.

Considering proposed concept of an object we can notice that the lack of the access by the algorithm of a given object Obj_r (by its function f_r^o) to the parameters $(\{x_{p_s}, x_{p_s+1}, ..., x_{k_s}\})$ of the object Obj_s is a too big restriction. To resolve the accessibility restriction problem a number of methods of access (under control) to the parameters of a given object has been proposed. The object Obj_s may make its own parameters accessible for the function fo_r of the object Obj_r (fig. 2).

During the development we introduce the concept of the object tools of controlled access to the internal parameters of another object. So, a given object Obj_r (its function f_r^o) has accessible its own parameters $\{x_{p_r}, x_{p_r+1}, ..., x_{k_r}\}$ called internal parameters of the object Obj_r, and in a limited way the parameters of other objects (for example to the parameters $\{x_{p_s}, x_{p_s+1}, ..., x_{k_s}\}$ of the object Obj_s. Obviously the object has access to the global parameters that are accessible to any object. There are tools to give control (restrictions) of the access to the internal parameters of an object. One of the most elegant tools is the use of methods defined in the body of the object to which internal parameters the access is realized. We may consider methods are tools for a particular communication between objects.

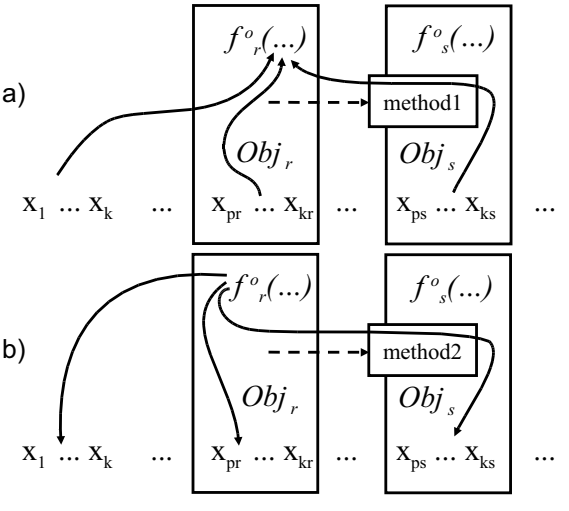


Fig. 2. Schema of the access of the object (partial algorithm) Obj_r to the parameters of the object Obj_s

The scenario of the access of one object (Obj_r) to parameters of another object (Obj_s) may be described as follow:

- Function f_r^o realizing execution of the object Obj_r needs the following information: the value of its own parameters of the object Obj_r, global parameters and the value of some parameters of the object Obj_s.
- The object Obj_r to get access to the parameters of the object Obj_s put in execution (by calling) a properly selected method of the object Obj_s and due to the execution of this method the object Obj_r obtain information about parameters of the object Obj_s. The access is controlled by the object Obj_s by the algorithm of its method and the access may be realized, partially realized or forbidden.
  We can consider the function f_r^o as following:

$$\begin{aligned}&(x'_1, ..., x'_k, x'_{p_r}, ..., x'_{k_r}, x'_{p_s}, ..., x'_{k_s})\\&= f_r^o(x_1, ..., x_k, x_{p_r}, ..., x_{k_r}, method1(x_{p_s}, ..., x_{k_s})).\end{aligned} \tag{15}$$

- When the object Obj_r has all the necessary information it can realize evolution of the state of the algorithm using its own function f_r^o. In the result of the execution the parameters of the algorithm may be modified. Object Obj_r may modify without restrictions its own parameters $(x_{p_r}, x_{p_r+1}, ..., x_{k_r})$ and global parameters. But sometimes as the result of application of the function f_r^o the modification of internal parameters of another (Obj_s) is necessary. For this purpose the methods of the object Obj_s may be used. Object Obj_r using the properly chosen method of the object Obj_s may change value of internal parameters of that object (for instance: $method2(x'_{p_s}, ..., x'_{k_s})$ (fig. 2)). Obviously this modification is controlled by the object Obj_s by algorithms of methods used.

The presented scenario of the communication between objects causes independency of an object to be considerably restricted. For instance in presented scenarios the object Obj_s may not give access to its internal parameters for the object Obj_r what may block algorithm of the object Obj_r. We can say that autonomy of the object Obj_r is depended (under restriction) on the object Obj_s (fig. 2).

Using the proposed concept of the object with tools of communication by the "call of method" we can decompose a given complex algorithm into partial algorithms represented by objects ([9]. Cooperation of this objects makes possible the execution of the complex algorithm as a whole. The realization of the complex algorithm is the result of the successive application of the functions f_r^o and f_s^o in a properly defined order (cooperation of the objects Obj_r and Obj_s).

4 Decomposition of an Algorithm with the Concept of an "Agent"

The loss of independency (or autonomy) of partial algorithms realized as objects is the price paid for the cooperation of objects in the system realized by the object oriented approach.

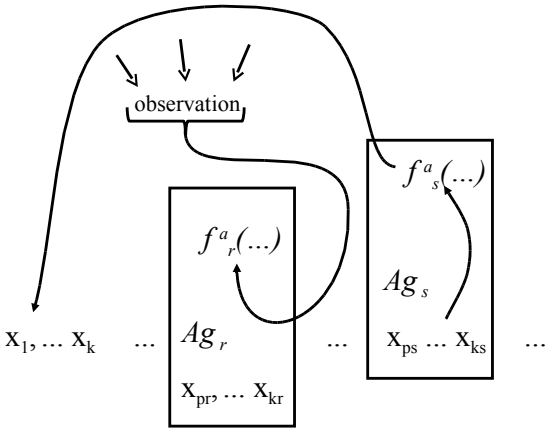


Fig. 3. Schema of execution of the algorithms of objects with the use of observation of behavior of other objects. It is a principle of the concept of an agent.

We can try to find a solution where partial algorithms are realized as entities more independent than objects. It is possible due to the observation operation. Using the observation operation one algorithm may observe behavior of another one. It leads us to the concept of an agent. The way of the action of a given agent may be illustrated by the following example:

- Function realizing algorithm of the agent Ag_r needs its own parameters, global data, and parameters of the agent Ag_s.
- Therefore the agent Ag_r observes behavior of the agent Ag_s. It means the agent Ag_r follows changes in the environment (global data) that are caused by the actions of the agent Ag_s. Due to this observation agent Ag_r may deduce the state of parameters of the agent Ag_s. This observation and deduction is an independent process of intentions of the agent Ag_s, although the received data do not give the full information about the internal parameters of the agent Ag_s. But information obtained may be sufficient for the agent Ag_r to continue its actions.
- Agent Ag_r being in possession of necessary information, and using its function f_r may modify its own parameters and state (global parameters) of the environment. These changes are realized usually as an execution of an event in the environment. Agent Ag_r has no possibilities to change directly the internal data of the agent Ag_s but by the means of modification of the state of the environment it have impact on the state of other agents. Agent Ag_s observes the changes in the environment caused by the agent Ag_r. Due to this changes agent Ag_s modifies its own parameters.

Agent observing environment (global data) and behavior of other agents may realize its actions (fig. 3). Presented concept of an agent gives considerable

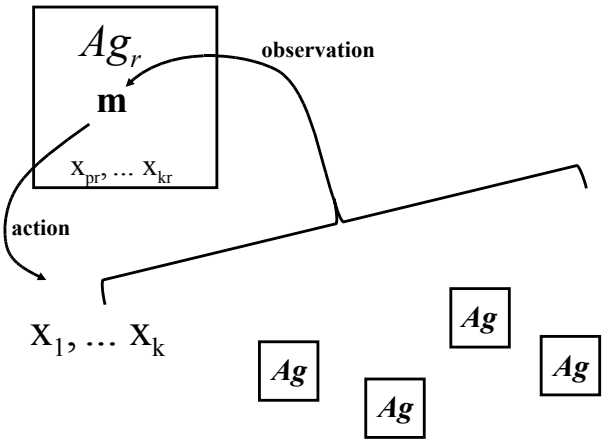


Fig. 4. Schema of an agent acting in the environment with the use of its model

augmentation of independence of an agent and possibility of interaction between agents. That is the crucial difference between the concepts of the object and the agent.

Resuming we can say that the source of information for the agent is the state of the global data (state of the environment, or resources of the environment represented by $(x_1, ..., x_k,))$, local data of the agent (state of the agent Ag_r represented by $(x_{pr}, ..., x_{kr},))$ and observed behavior of other agents (for instance Ag_s). This information agent may acquire by observation of the environment(fig. 4).

The algorithm is created as the environment containing resources (represented by global data) and agents that remain and act in the environment.

Not all information acquired by the observation of the environment is necessary for a given agent to be able to act in the environment. To select the only necessary information agent may use a method of modeling the environment. Due to the operation of observation a given agent builds in its mind a model of the environment. Using this model m agent may select the necessary information, and then use it to realize its action in the environment. This selection is realized by the adopted abstraction level of the model construction and its manipulation.

Using presented idea we can consider the following scenario of action by a given agent that is in a given environment:

- Agent using the operation of observation acquires information from the environment and builds a model (m) of the environment in its mind. The model is created with adopted by the agent abstraction level - what means that in the model only some characteristic properties of the environment are memorized (only information necessary for agents's actions).
- Agent plans actions in the environment using the analysis of created model m.
- Agent carries out the planned actions realizing events in the environment.

Agent Ag_r to observe the changes in the environment caused by the agent Ag_s has to memorize in the model m not only actual state of the environment $(x_1^i, ..., x_k^i,)$ but also ,,historical data" $(x_1^{i-1}, ..., x_k^{i-1},)$describing the state of the environment before changes caused by the agent Ag_s. That is the model m of the agent that contains all data (including "historical data") necessary for the agent to act in the environment. The application of the model m may enable the autonomy of the agent and realize cooperation among agents.

5 Conclusion

Presented approach to the description of decomposition of the algorithm helps to define the concept of the object and the agent.

The paper makes possible to explain reasons for introduction of the object and agent notions. It enables to determine the principal difference between an object and an agent and to establish their main properties.

It may be useful for development of Object Oriented Technologies and Agent Oriented Technologies to create complex multiagent systems.

References

1. Cetnarowicz, E., Cetnarowicz, K., Nawarecki, E.: The simulation of the behavoiur of the world of autonomous agents. In: Proc. of the XVII International Czech - Poland - Slovak Colloqium - Workshop: Advanced Simulation of Systems, Zabrzech na Morave, Czech Republic, vol. 2, pp. 107–115 (1995) ISBN 80-901751-4-7
2. Cetnarowicz, K.: Problems of the evolutionary development of the multi-agent world. In: Proc. of the First International Workshop: Decentralized Multi-Agent Systems DIMAS 1995, Krakow, Poland, pp. 113–123 (1995) ISBN 83-86813-10-5
3. Cetnarowicz, K., Cetnarowicz, E.: Multi-agent decentralised system of medical help. In: Management and Control of Production and Logistics. IFIP, IFAC, IEEE Conference, Grenoble, France, 2000. ENSIEG, LAG Grenoble, France (2000)
4. Cetnarowicz, K., Dobrowolski, G., Ko@zlak, J.: Active agents cooperation in decentralized systems. In: Bubnicki, Z. (ed.) Proc. of the 12th Int. Conf. on Systems Science, vol. 1, pp. 57–62. Oficyna Wydawnicza Politechniki Wroc@lawskiej (1995) ISBN 83-7085-152-5
5. Cetnarowicz, K., Nawareck, E.: Système d'exploitation decentralisé realisé à l'aide de systèmes multi-agents (Operating System Realized with the Use of Multi-agent Systems). In: Proceedings, Troisième Journées Francophone sur l'Intelligence Artificielle Distribuée et les Systèmes Multiagents, St Baldoph, Savoie, Francja, pp. 311–322 (1995)
6. Crowley, J.L., Demazeau, Y.: Principles and techniques for sensor data fusion. In: Signal Processing, vol. 32, pp. 5–27. Elsevier Science Publishers B. V., Amsterdam (1993)
7. Nawarecki, E., Cetnarowicz, K.: A concept of the decentralized multi-agent rt system. In: Proc. of the International Conference Real Time 1995, Technical University of Ostrava VSB, Ostrava, Czech Republic, pp. 167–171 (1995) ISBN 80-901751-6-3

8. Nawarecki, E., Cetnarowicz, K., Cetnarowicz, E., Dobrowolski, G.: Active agent idea applied to the decentralized intelligent systems development. In: Štefan, J. (ed.) Modeling and Simulation of Systems MOSIS 1994, Ostrava, pp. 64–71. House of Technology Ltd. (1994) ISBN 80-901229-8-1
9. Nicola, J., Coad, P.: Object-Oriented Programming. Prentice Hall, Inc., Englewood (1993)
10. Rao, A.S., Georgeff, M.P.: Modelling rational agents within a bdi architecture. In: Proc. of the Second International Conference on Principles of Knowledge Representation and Reasoning, KR 1991, Cambridge, MA, USA, pp. 473–484 (1991)
11. Shoham, Y.: Agent-oriented programming. Artificial Intelligence 60, 51–92 (1993)
12. Weiss, G.: Multiagent Systems. MIT Press, Cambridge (1999)
13. Winkowski, J.: Programowanie symulacji procesow (Programming of Simulation Process). Wydawnictwo Naukowo-Techniczne, Warszawa (1974)
14. Wirth, N.: Algorithms+Data Structures = Programs. Prentice-Hall Series in Automatic Computation (1976)

The Norm Game - How a Norm Fails

Antoni Dydejczyk, Krzysztof Kułakowski, and Marcin Rybak

Faculty of Physics and Applied Computer Science,
AGH University of Science and Technology,
al. Mickiewicza 30, PL-30059 Kraków, Poland
dydejczyk@ftj.agh.edu.pl,
kulakowski@novell.ftj.agh.edu.pl,
fisher@autocom.pl

Abstract. We discuss the simulations of the norm game between players at nodes of a directed random network. The final boldness, i.e. the probability of norm breaking by the players, can vary sharply with the initial boldness, jumping from zero to one at some critical value. One of the conditions of this behaviour is that the player who does not punish automatically becomes a defector. The threshold value of the initial boldness can be interpreted as a norm strength. It increases with the punishment and decreases with its cost. Surprisingly, it also decreases with the number of potential punishers. The numerical results are discussed in the context of the statistical data on crimes in Northern Ireland and New Zealand, on divorces in USA, and on the alcohol consumption in Poland.

1 Introduction

The cultural paradigm of the supremacy of an individual over collective thought is well established in our culture. On the contrary to this, there is much evidence that individual decisions and beliefs strongly depend on the social environment. The effects related are the spiral of silence [1,2], the pluralistic ignorance [3,4], the bystander effect [5], the group polarization [6], the groupthink [7], the bandwagon effect [8,9], the Abilene paradox [10], stereotyping [11,12] and others. All these effects show that social groups influence or sometimes even determine their members' behaviour. It is natural that this notion shifts attention of scientists to investigations of an interaction between players rather than the players themselves, notwithstanding this interaction perception merely in individual minds. A general review of applications of statistical physics in player-based modeling can be found in [13]. A physicist could be surprised how far the analogy can be driven between the cognitive-emotional social field [14] and the electromagnetic field [15].

The subject of social norms seems to belong to these parts of the science on human behaviour, which can be treated as a systematic science (in terms of [14]). Up to our knowledge, the first attempt of such a treatment was done by Robert Axelrod [16,17]. A brief introduction to the problem can be found in the introduction to [18]. Axelrod was inspired by the idea of the evolutionary principle

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 835–844, 2009.

(EP), formulated by himself as follows: *...what works well for a player is more likely to be used again, whereas what turns out poorly is more likely to be discarded.* [19]. Axelrod enumerated three independent mechanisms which lead to the realization of EP: *i)* more effective individuals survive more likely, *ii)* players learn by trial and error and maintain strategies which are more effective, *iii)* players observe each other and imitate the winners. Axelrod suggested that in the problem of norms, the third mechanism is most applicable. In the opinion of the present author the second mechanism is even more important for norms which can persist in the timescale of the order of human life. The argument is that such norms are characteristic for a social group rather than for a given person, and they are obeyed sometimes just by tradition. Investigation of this kind of norms with statistical tools makes sense, as the psychics of an individual player can be reduced there to a black box. This is the area where a sociophysicist is allowed to enter. For a review on games on graphs see [20].

In our previous work [21] a new scheme of simulation was formulated for the norm game [16,17]. Most briefly, the original version of the norm game can be described as follows. N players are positioned at nodes of a fully connected network. The players are endowed with their initial boldnesses, i.e. probabilities of defecting (breaking the norm), and with initial vengeances, i.e. the probabilities of punishing the defection, if observed at other players. Next, a player is selected and tempted to break the norm. If he does, he is granted with some prize. Further, if a neigbour of the defector decides to punish, the defector has to pay a fine. Finally, also the punisher incurs some cost of his action. An application of the genetic algorithm leads to spreading of most effective strategies.

This game, with some simplifications described in Sect. II, was simulated recently [21] in a directed Erdös-Rényi network [22,23]. The main result was the sharp dependence of the final distribution of the players' boldness on their initial boldness. The values of the initial boldness of the players were selected from the range $(0.9\rho, \rho)$, where $0 < \rho < 1$. Then, ρ is a measure of the initial boldness in the system: initially, all players defect with probability close to ρ. Below some threshold value ρ_c of ρ, the mean value of the final boldness was found to be close to zero. When $\rho > \rho_c$, the same mean value was found to be close to one. Calculated values of ρ_c were found to decrease with the cost parameter γ. In parallel, a similar bistability was obtained by means of analytical calculations [21,24].

The aim of the present work is to investigate this discontinuity. New numerical results to be reported here are as follows:

i) A necessary condition for the appearance of the discontinuity is that once a player abstains from punishing, his boldness is set to one.

ii) The discontinuity remains in the undirected growing networks: both in the scale free and the exponential ones, but it disappears in the directed growing networks.

iii) The value of ρ_c increases with the punishment and it decreases with the mean node degree.

These results are described in Sect. III with more details. They are discussed in Sect. IV in the context of some statistical data, which display a relatively quick variations in time. Final remarks in Sect. V close the text.

2 The Simulation Algorithm

Basically, we use the same algorithm as in [21] for different kinds of networks. Players are placed at nodes of a network of a given topology. For the directed Erdös-Rényi network, for each node i its degree $k(i)$ is selected from the Poisson distribution, and then the neigbors of i are chosen with uniform probability. The links are directed from the neigbours to the node k; this means that the player at i can be punished by his neighbours, but not the opposite. Obviously, each neigbour has its own neighbours who can punish him.

As the next step, each player i is endowed by the probability that he defects $b(i)$ (boldness) and the probability that he punishes $v(i)$ (vengeance), with the condition $b(i)+v(i) \leq 1$. The boldnesses are selected from the range $(\rho-0.01, \rho)$, where ρ is a real number from the range $(0.01, 1)$. As before, ρ is a measure of the initial boldness in the system. For each i, the vengeance is set as $v(i) = (1-b(i))/\mu$, where $\mu \geq 1$. Then, the initial state is described by two parameters ρ and μ. As a rule, $\mu = 1$ or $\mu = 2$.

The dynamics is described by two parameters β and γ. A node i is selected randomly. The player at i defects with probability $b(i)$. If he does, his boldness $b(i)$ is set to 1, and his vengeance $v(i)$ is set to 0. If he does not, $b(i)$ is set to 0 and $v(i)$ is set to 1. This settings can be treated as an example of social labelling [27]. Then, a neighbour j of i is checked if he punishes or not, with his actual vengeance $v(i)$. If he does not punish, he is treated as a defector. Then, his boldness $b(j)$ is set to 1 and his vengeance $v(j)$ is set to 0. If j punishes, the boldness of punished i is reduced by a factor $1-\beta$, i.e.

$$b(i) \rightarrow b(i)(1-\beta) \tag{1}$$

Simultaneously, the punisher j pays the cost of punishing, what leads to a reduction of his vengeance as

$$v(j) \rightarrow v(j)(1-\gamma) \tag{2}$$

In this way, we omit the parameters of the payoffs, which are not necessary.

Once the defector is punished, the algorithm selects another player i to be tempted to defect. The simulation is repeated for the undirected growing networks: exponential and scale-free ones [25,26], and for the directed growing networks. In the last case the direction of nodes was selected either from the new node to its older neigbours, or the opposite.

3 Numerical Results

Although the threshold effect is present for the directed Erdös-Rényi networks [21], it disappears in the directed Albert-Barabási networks where the direction of edges

is chosen from newly attached nodes to their older neighbours. The effect disappears also in the latter case when the direction of all edges is inverted. These results are shown in Fig. 1. Then, this combination of the age of nodes and the

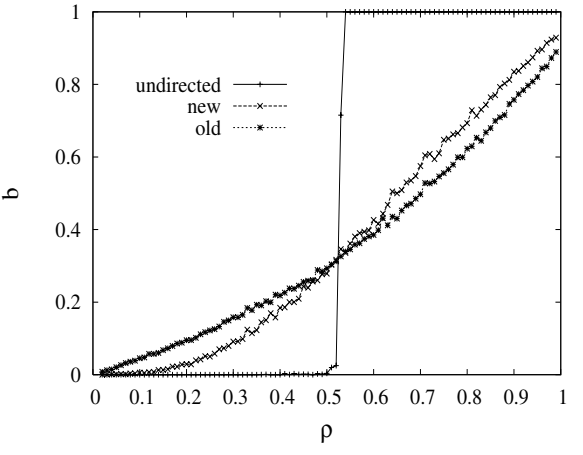
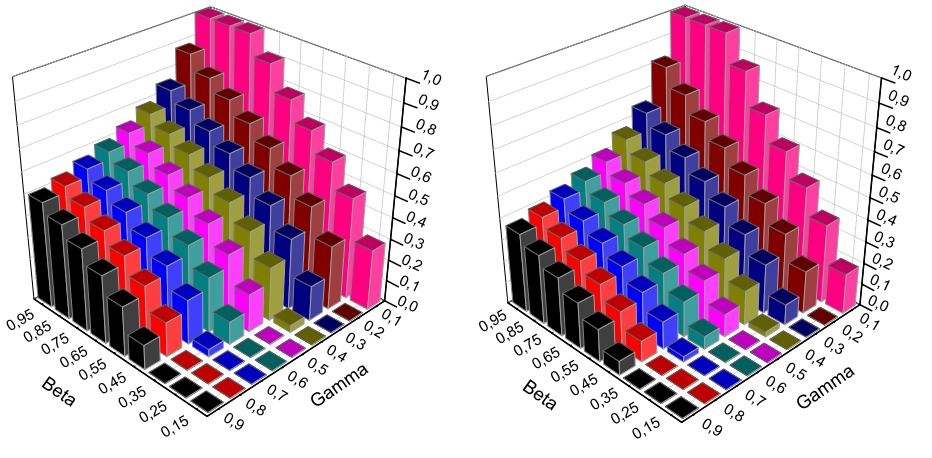


Fig. 1. The mean boldness for Albert-Barabási growing networks with $M = 3$. For the undirected network the threshold value is observed at ρ_c about 53. For two directed networks the direction is either from new to old nodes or the opposite. Then, either new nodes can punish those to which they are attached (the plot 'new'), or the old nodes can punish those attached to them (the plot 'old'). No threshold is observed for the directed networks.



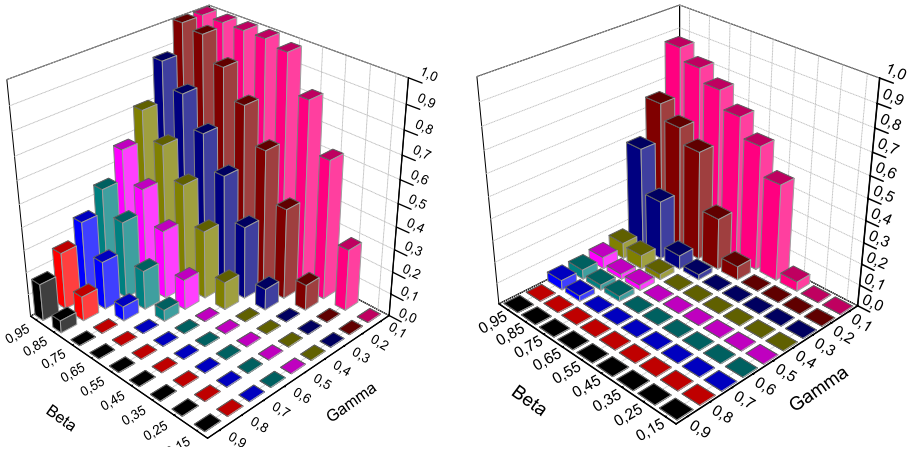
(a) $\mu = 1$ (b) $\mu = 2$

Fig. 2. The value of the norm strength ρ_c for the directed Erdös-Rényi network, for $\mu = 1$ and $\mu = 2$. The system size is 10^4 nodes. The mean in-degree $< k >= 2$.

direction of edges between them does modify significantly the system behaviour. The nature of this modification remains to be clarified.

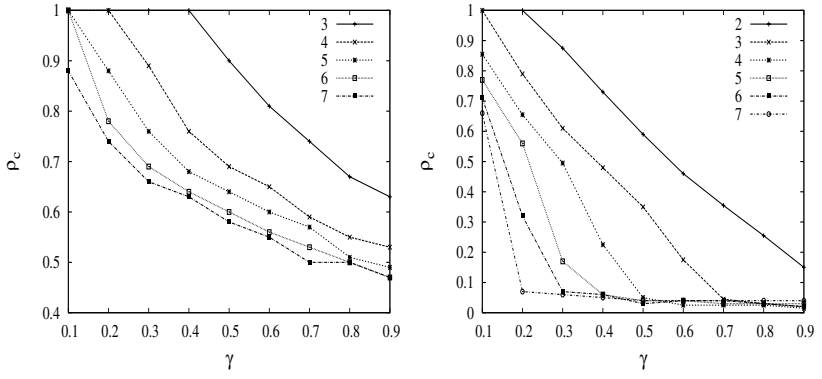
The simulations were concentrated on the calculation on the threshold boldness ρ_c against the punishment parameter β and the cost parameter γ. Let us repeat that if the initial value of ρ, what is a rough measure of the initial boldness, is smaller than ρ_c, then the final boldness is small; the defection is rare. On the contrary, if $\rho > \rho_c$, most players defect in the stationary state. In other words, it is more difficult to defect if ρ_c is large. Then, ρ_c can be treated as a measure of the norm strength. The possible size effect was checked for an example of ten directed Erdös-Rényi networks of size from $N = 10^3$ to 1.5×10^4. The threshold value ρ_c was 0.41 for $N = 1000$ and remained 0.40 for all higher N. Its root mean square deviation decreased with N from 0.016 to 0.005. Also, the time to get the final value of the mean boldness remained about 40 timesteps for all above values of N, where one timestep is equivalent to a temptation of N randomly selected players. The simulations confirm the previous result [21] that ρ_c increases with the punishment constant β. Also, they give a new result, that ρ_c decreases with the punishment cost γ. Both facts are natural; the norm is stronger if the punishment is heavy for the defector, and it is weaker if the punishment is costful for the punisher. These results are shown in Fig. 2 for the directed Erdös-Rényi network with the mean number of potential punishers $\lambda = 5$, and in Fig. 3 for the undirected Albert-Barabási networks.

On the other hand, these results suggest that the norm strength decreases with the potential number of neighbours. To verify this, we have calculated ρ_c for the directed Erdös-Rényi networks for various numbers of in-neigbours λ as well for the undirected scale-free networks for various numbers of the parameter M. The results are shown in Fig. 4 a,b. In both cases the same effect is observed, that the



(a) $M = 2$ (b) $M = 4$

Fig. 3. The value of the norm strength ρ_c for the undirected scale-free Barabási-Albert network, for $M = 2$ and $M = 4$, $\mu = 2$. The system size is 10^4 nodes.



(a) directed ER (b) undirected AB

Fig. 4. The value of the norm strength ρ_c for the directed Erdös-Rényi networks for $\lambda = 3 - 7$, $\beta = 0.95$, $\mu = 2$ and for the undirected scale-free Albert-Barabási network, for $M = 2 - 7$, $\beta = 0.95$, $\mu = 2$

norm strength ρ_c decreases with the mean number of potential punishers, which is λ for the the directed Erdös-Rényi networks and $2M$ for the undirected scale-free networks. This results is somewhat surprising. Before the calculation is done, we expected that an increase of the number of punisher should make the norm more strong. We have to admit that the process contains some internal feedback from those tempted to their neighbours. Once the norm is broken, those who can punish have to decide if the punishment cost is not too large. According to our algorithm once they do not punish, they become defectors. If the norm was broken at all, this outcome was not possible. In this sense, the larger number of potential punishers who do not punish can lead to an avalanche of new defectors. This is a possible explanation of the observed decrease of ρ_c with the number of neighbors.

4 Some Stylized Facts

Some statistical data suggest that indeed some 'equilibrium' social states are possible, where a norm is obeyed to some definite extent. Once such a state ceases its stability, the system moves to another state, where the mean boldness is constant again or it fluctuates only slowly. This transformation can be seen in the statistical data as a large change of a given quantity, accompanied by its relatively slow variation before and after the change. Such a change can be interpreted as a discontinuous jump between two states. Here we are going to mention four examples and to comment them shortly.

Our first example is the number of recorded criminal offences per year in Northern Ireland. Between 1980 and 1997 it varied between 50 thousands and 70 thousands, but it increased to about 120 thousands in 1997 [28,29]. This jump was probably triggered by the fact that since 1997 the Provisional IRA

has observed a ceasefire. At this moment, large hidden arsenals became useless for the patriotic purposes. As a consequence, the social norm 'preserve your gun for the outbreak' has failed. The data on murders in New Zealand between 1950 and now show a similar dynamics [30]. Before 1986 the number of homicides did not excess 60 per year, but it was not smaller than 120 per year between 1987 and 1994. This change is correlated with the transformation of New Zealand to free-trade economy, started in 1984. Here the norm is the Commandment 'You shall not murder' which did not cease its universality in 80-s; perhaps it was the punishment what decreased. Our third example is the number of divorces in USA. The data [31] show that this number increased twice (from about 2.5 to about 5.0 per 1000 population) between 1965 and 1975. This change was accompanied by a similarly large relative increase of numbers of rapes, murders and robberies within the same time period [32]. Obviously, this fall of norms was triggered by the Vietnam war, but it is not easy to separate this cause of divorces from the accompanying sexual revolution. Divorced people often look for other partners, not necessarily unmarried, what accelerates the process like in a chain reaction [33]; still the related lifetime is not shorter than a couple of years. Our last example is the fall of spirits consumption in Poland in 1981 from 8 to 6 litres of pure alcohol per an adult per year [35]. At that time, Polish people had drank mainly spirits, while the French preferred wine and the German - beer. To drink alcohol is in Poland a kind of social norm, to show friendship and empathy, paying no heed to health losses. This need for a demonstration of common feeling of being unhappy was released by the new hope of a political transformation, brought by the free trade union "Solidarity" and the message of John Paul II. The political transformation in Europe is not visible at the same data for other countries; the consumption of wine in France and of beer in Germany decreased rather smoothly [35]. The plots on the statistical data on all of four examples are shown in initial and somewhat more extended version of this paper in arXiv [34].

As we see, all these changes were triggered by some events: the ceasefire in Northern Ireland, the economic transformation in New Zealand, Vietnam war in USA or political reforms in Poland. To interpret these changes in terms of our model picture, we admit that in each case the event as those listed above changes the punishment or/and it cost. This influences the temptation process, which is more or less steady in the society; new and new people are faced with the question: to divorce or not, to drink or not, to break law or not. Their decisions are influenced by individual attitudes of their neighbours, which are also determined in the process. As an output we see a social change in macroscopic data. We note that according to our numerical results, the abrupt fall of norm preserving is a consequence of the rule 'those who abstain from punishing, defect'. In other words, the society is divided into defectors and punishers. A question arises, if this rule makes sense in our four examples. To find an answer, we have to identify the punishment. In the first and second example to punish is to report the case to the police. Then, the interpretation is clear: who did not report to the police is guilty. In the third example to punish could be to condemn divorces in public. However, in principle

one could be faithful in marriage and tolerant towards manifestations of sexual promiscuity. We face the same problem also in the fourth example, where many people prefer the strategy 'don't drink but let drink'. Then, our numerical result seems to apply to two examples out of four. This difficulty is solved, if the norm itself is redefined as 'the duty to punish'. According to this new point of view, to obey a norm means to declare in public that this norm should be obeyed. In particular in our third example what is important is to declare in public that the institution of marriage is holy and inviolable. Similarly in the fourth example the norm is to insist that to be accepted in the group, everybody must drink. In this way, the metagame (in terms of Axelrod) is the game. This transformation can be interpreted theoretically as follows: the preservation of a given norm should be evaluated by counting not those people who preserve it, but rather those who punish for its defection. This formulation raises the question about the status of norms which vanish in a continuous way. This in turn calls for an experimental criterion, which change is continuous and which is sharp. These questions are beyond the scope of this paper.

5 Final Remarks

Even if numerically complete, the study is preliminary. It reveals the connection between consequences of decisions of individual players and the state of the society at a macro-level. In the above calculations, these consequences are chosen as to label the players: who once defected will never punish and who once punished will never defect. In this way, the society is divided into two opposite groups. These rules of labelization should be justified and relaxed if necessary in accordance with the social reality, and these modifications in each case do depend on the specific norm under consideration.

The discontinuity found here for the norm game is similar to the sharp variation of a collective behaviour, described in [36] within a dynamic model of a social imitation. However, the mechanism is different; what is collective in our model of the norm game is the social labeling, when decisions of the players, triggered by a single defection, classify them irrevocably as bold or vengeful. On the contrary to the results of [36], the time variation of the boldness and vengefulness display no hysteresis effect. However it is possible that the norm game can be described within the Random Field Ising Model, as applied in [36]. Indeed, an appropriate reformulation of this formalism could provide an insight to the time dynamics of the social behaviour with respect to norms.

To conclude, the sharp dependence of the final boldness on its initial distribution follows from the labelling consequences of initial decisions of the players. This sharpness does not depend much on the network topology. However, it can be removed if the direction of the links is determined by the age of nodes in growing networks.

Acknowledgements. We thank Jean-Philippe Bouchaud for pointing us reference [36]. The research is partially supported within the FP7 project SOCIONICAL, No. 231288.

References

1. Noelle-Neumann, E.: The spiral of silence: A theory of public opinion. Journal of Communication 24, 43–51 (1974)
2. Noelle-Neumann, E.: The Spiral of Silence: Public Opinion - Our Social Skin. University of Chicago Press, Chicago (1993)
3. Allport, F.H.: Institutional Behavior. University of North Carolina Press, Chapel Hill (1933)
4. Prentice, D.A., Miller, D.T.: Pluralistic ignorance and alcohol use on campus: Some consequences of misperceiving the social norm. Journal of Personality and Social Psychology 64, 243–256 (1993)
5. Latané, B., Darley, J.M.: The Unresponsive Bystander: Why Doesn't He Help? Prentice Hall, Englewood Cliffs (1970)
6. Moscovici, S., Doise, W.: Conflict and Consensus: A General Theory of Collective Decisions. Sage Publ., London (1994)
7. Janis, I.L., Mann, L.: Decision making: A psychological analysis of conflict, choice, and commitment. Free Press, NY (1977)
8. Navazio, R.: An experimental approach to bandwagon research. Public Opinion Quarterly 41, 217–225 (1977)
9. Morowitz, V.G., Pluziński, C.: Do polls reflect opinions or do opinions reflect polls? The impact of political polling on voters' expectations, preference, and behavior. J. of Consumer Research 23, 53–67 (1996)
10. Harvey, J.B.: The Abilene Paradox. Jossey-Bass, San Francisco (1996)
11. Lippmann, W.: Public Opinion. Free Press Paperbacks, NY (1997) (first edition NY 1922)
12. Haslam, S.A., Salvatore, J., Kessler, T., Reicher, S.D.: How Stereotyping Yourself Contributes to Your Success (or Failure). Sci. Am. (April 2008)
13. Castellano, C., Fortunato, S., Loreto, V.: Statistical physics of social dynamics (arXiv:0710.3256)
14. Brown, J.F.: Individual, Group, and Social Field. American J. of Sociology 44, 858–867 (1939)
15. Sallach, D.L.: Social Field Theory: Concept and Application (preprint 2006)
16. Axelrod, R.: An evolutionary aproach to norms. Amer. Political Sci. Rev. 80, 1095–1111 (1986)
17. Axelrod, R.: Complexity of Cooperation player-Based Models of Competition and Collaboration. Princeton University Press, Princeton (1997)
18. Fent, T., Groeber, P., Schweitzer, F.: Coexistence of social norms based on in- and out-group interaction. Advances in Complex Systems 10, 271–286 (2007)
19. Axelrod, R.: The Evolution of Cooperation. Basic Books, NY (1984)
20. Szabó, G., Fáth, G.: Evolutionary games on graphs. Phys. Reports 446, 97–216 (2007)
21. Kułakowski, K.: Cops or robbers - a bistable society. Int. J. Mod. Phys. C 19, 1105–1111 (2008)
22. Bollobás, B.: Random Graphs. Cambridge UP, Cambridge (2001)
23. Caldarelli, G.: Scale-Free Networks. Oxford UP, Oxford (2007)
24. Kułakowski, K.: The norm game in a mean-field society. Journal of Economic Interaction and Coordination (in print) (arXiv:0801.3520)
25. Bornholdt, S., Schuster, H.G. (eds.): Handbook of Graphs and Networks: From the Genome to the Internet. Wiley-VCH, Berlin (2003)

26. Dorogovtsev, S.N., Goltsev, A.V.: Critical phenomena in complex networks. Rev. Mod. Phys. 80, 1275–1335 (2008)
27. Marshall, G. (ed.): The Concise Oxford Dictionary of Sociology. Oxford UP, Oxford (1998)
28. data of Eurostat, http://epp.eurostat.ec.europa.eu/
29. French, B., Freel, R.: Experience of crime: Findings from the 2005 Northern Ireland crime survey, Northern Ireland Office, Research and Statistical Bulletin (2/2007), http://www.equality.nisra.gov.uk/Experiencesofcrime2005crimesurvey.pdf
30. data of the Sensible Sentencing Trust, http://www.safe-nz.org.nz/statsgraph.htm
31. Monthly Vital Statistics Report 39(12) suppl. 2, May 21 (1991)
32. data of the Disaster Center, http://www.disastercenter.com/crime/uscrime.htm
33. Everett, C.: Legal Decisions and Family Outcomes. Routledge, London (1998)
34. Kulakowski, K., Dydejczyk, A.: arXiv:0810.5291
35. data of the Food and Agriculture Organization of the United Nations, World Drink Trends (2003)
36. Michard, Q., Bouchaud, J.-P.: Theory of collective opinion shifts: from smoooth trends to abrupt swings. Eur. Phys. J. B 47, 151–159 (2005)

Graph Grammar Based Petri Nets Model of Concurrency for Self-adaptive *hp*-Finite Element Method with Triangular Elements

Arkadiusz Szymczak and Maciej Paszyński

Department of Computer Science
AGH University of Science and Technology,
Al. Mickiewicza 30, 30-059 Cracow, Poland
arek.szymczak@gmail.com, paszynsk@agh.edu.pl
http://home.agh.edu.pl/~paszynsk

Abstract. The paper presents the model of concurrency for the self-adaptive *hp*-Finite Element Method (*hp*-FEM) with triangular elements. The model concerns the process of an initial mesh generation as well as mesh adaptation. The model is obtained by defining CP-graph grammar productions as basic undivided tasks for both mesh generation and adaptation algorithms. The order of execution of graph grammar productions is set by control diagrams. Finally, the Petri nets are created based on the control diagrams. The self-adaptive *hp*-FEM algorithm modeled as a Petri net can be analyzed for deadlocks, starvation or infinite execution.

1 Introduction

In this paper we present the model of concurrency for the self-adaptive *hp*-Finite Element Method (*hp*-FEM) with triangular elements. The self-adaptive *hp*-FEM algorithm starts from an arbitrary initial mesh and generates a sequence of meshes delivering exponential convergence of the numerical error with respect to the mesh size [4,5]. This is done by breaking selected elements into smaller elements (this procedure is called h refinements) or by modifying the polynomial order of approximation on selected element edges or interiors (this procedure is called p refinement). The algorithm can be utilized for solving two and three dimensional engineering problems with very high accuracy [4,5].

The parallel version of both two and three dimensional self-adaptive *hp*-FEM algorithms were created according to the domain decomposition paradigm [6,7]. The parallel implementations, although efficient and scalable, showed the need for further theoretical analysis of the concurrency hidden within the self-adaptive algorithm. Two graph grammar models for both sequential and parallel *hp*-FEM algorithms were created, for both rectangular [8] and triangular [9] finite elements. The models utilize the CP graph grammar introduced by [1,2,3].

In this paper, we extend the graph grammar model presented in [9] for the case of generation of an arbitrary initial mesh. We also create the model of concurrency, expressed by control diagrams setting the order of execution of

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 845–854, 2009.

graph grammar productions, and presenting which productions can be executed concurrently. Finally, the Petri nets are created, for both initial mesh generation and mesh adaptation algorithms. The Petri nets model will allow for creating the reachability graphs, finding place/transition invariants, and similar tools for the self-adaptive *hp*-FEM algorithm. The *hp*-FEM modeled as a Petri net can be analyzed for deadlocks, starvation or infinite execution.

2 2D Mesh Generation

In this section we present a set of graph grammar productions for generation of two-dimensional mesh with triangular finite elements. This extends the graph grammar introduced in [9] generating a linear sequence of triangular elements. The new productions **PI1** - **PI8**, presented in Figure 1, realize the topology generation of an arbitrary shape 2D triangular mesh. Meaning of the node symbols is as follows:

– *E1, E2* - type-1 (triangle) and type-2 (upside-down triangle) element, respectively

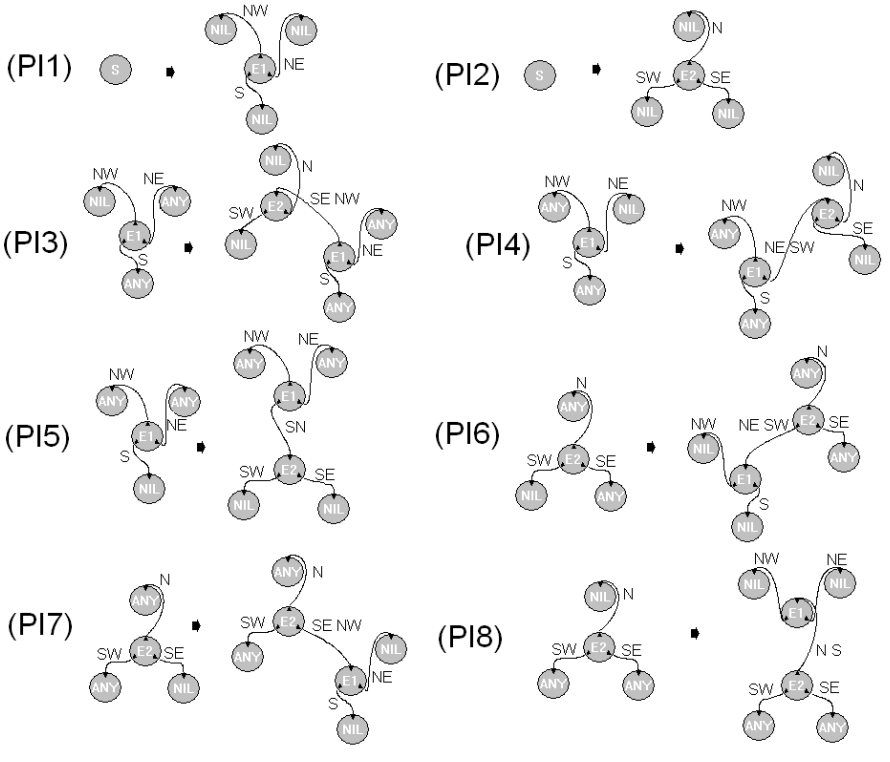


Fig. 1. Set of graph grammar productions **PI1** - **PI8** for the generations of triangular finite element two dimensional mesh

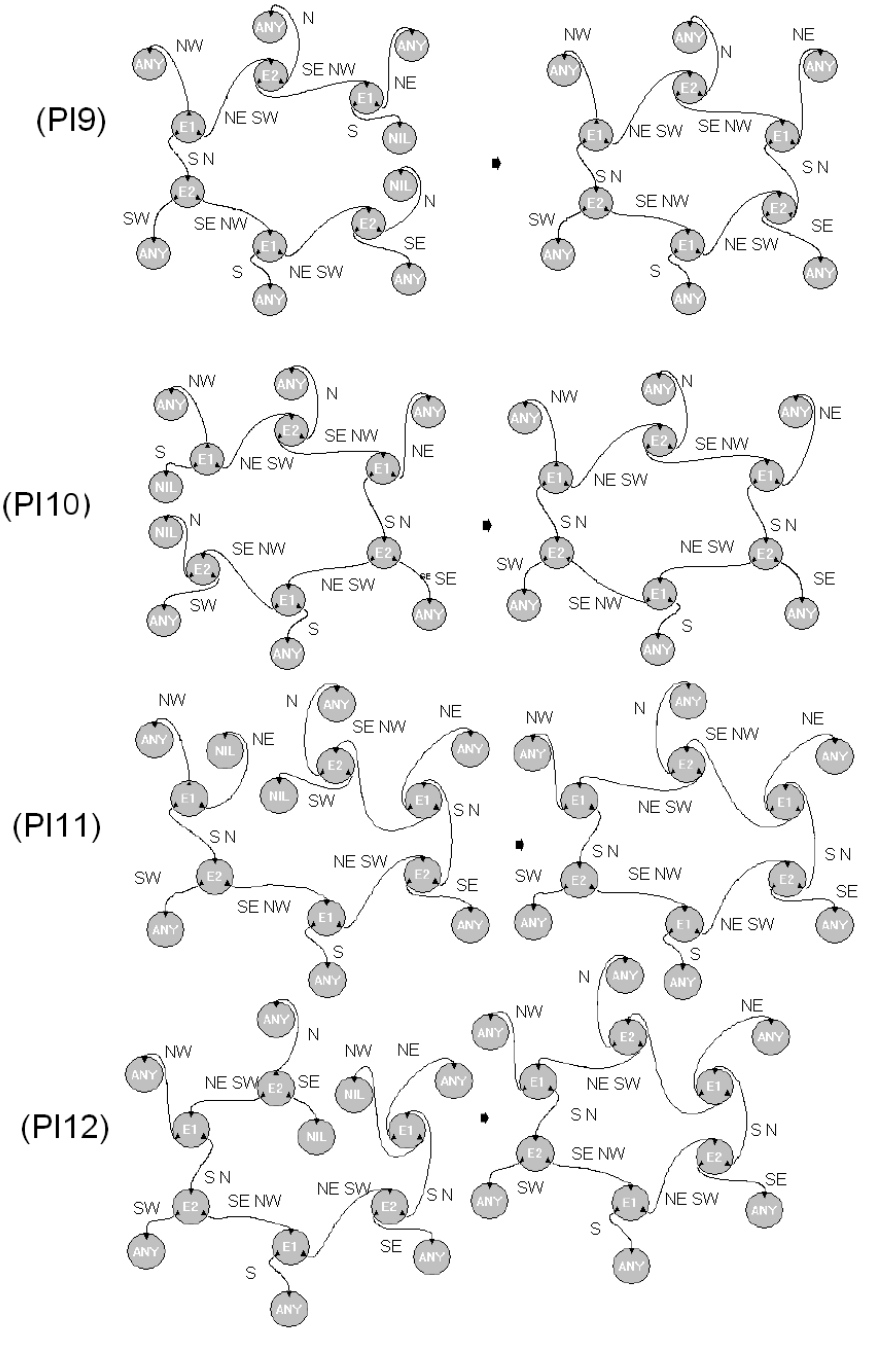


Fig. 2. Graph grammar productions **PI9 - PI12** connecting layers of generated elements

– *NIL* - fake node denoting no adjacent node
– *ANY* - fake node meaning that given node is irrelevant to the given production

The new productions **PI9** - **PI14**, presented in Figure 2 and 3, are responsible for identification of adjacent elements to prevent element overlapping. Since new element is always derived from one existing element and each element can have up to 3 adjacent elements, missing up to 2 bonds must be detected. Therefore, as soon as any of productions **PI9** - **PI14** matches, it must be executed before subsequent execution of productions **PI1** - **PI8**.

We summarize this section with an exemplary derivation of a single row of triangular finite element mesh. The mesh is generated by the following sequence of graph grammar productions: **PI1** - **PI4** - **PI7** - **PI4**. The resulting graph and the corresponding finite element mesh is presented in Figure 4.

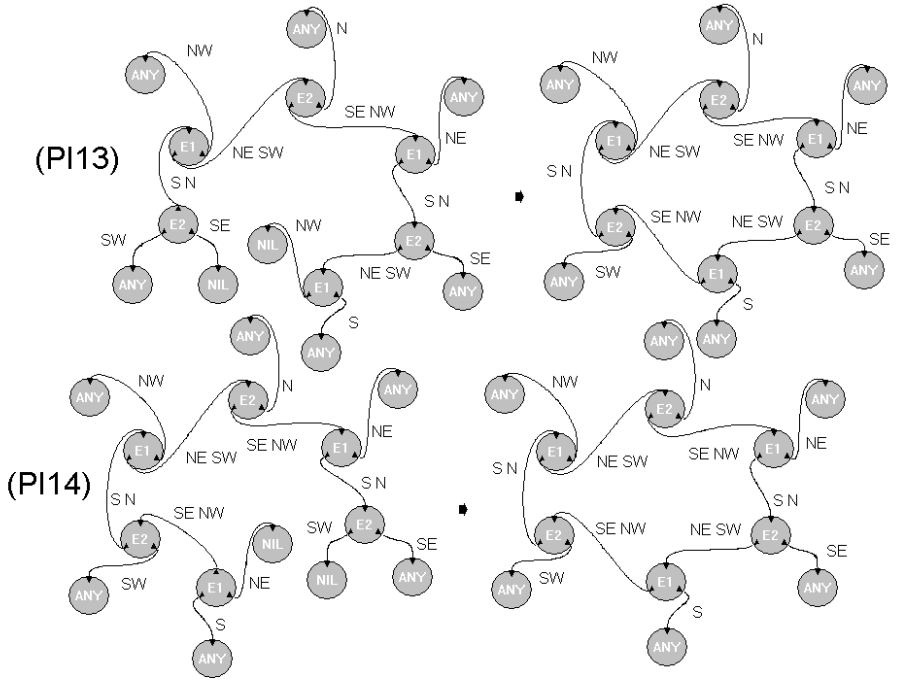


Fig. 3. Graph grammar productions **PI13** - **PI14** connecting layers of generated elements

3 Graph Grammar Based Model of Concurrency

In this section we present the control diagram based model of concurrency for the process of an initial mesh generation as well as mesh h adaptation. The model

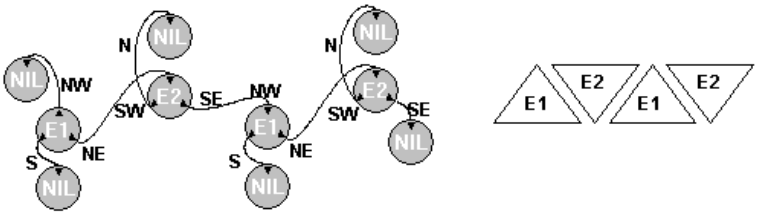


Fig. 4. Exemplary finite element mesh obtained by executing the following sequence of graph grammar productions **PI1** - **PI4** - **PI7** - **PI4**

is presented in Figure 5. The following list summarized symbols used in the control diagrams. Some graph grammar productions referred in the list were defined in [8], while the productions **PI1** - **PI14** were described in the previous section.

- *PE1* - production for generation of type 1 element structure
- *PE2* - production for generation of type 2 element structure
- *PCEH* - production for identification of common edge of horizontally adjacent elements
- *PCEV* - production for identification of common edge of vertically adjacent elements
- *PJI1* - production allowing for breaking type 1 element interior
- *PJI2* - production allowing for breaking type 2 element interior
- *PFE* - production allowing for breaking edge between two broken interiors
- *PBI1* - production for breaking type 1 element interior
- *PBI2* - production for breaking type 2 element interior
- *PBE* - production for breaking element edge
- *PWest1* - production for propagation of adjacency data for west edge of element type 1
- *PEast1* - production for propagation of adjacency data for east edge of element type 1
- *PSouth* - production for propagation of adjacency data for south edge of element type 1
- *PWest2* - production for propagation of adjacency data for west edge of element type 2
- *PNorth* - production for propagation of adjacency data for north edge of element type 2
- *PEast2* - production for propagation of adjacency data for east edge of element type 2

The generation of an initial mesh can be highly parallelized, as it is illustrated on the left panel in Figure 5. While subsequent elements are being added to the mesh, the structure of already existing elements may be constructed and their common edges may be identified. Adaptation of the mesh is more sequential, as it is presented on the right panel in Figure 5. Generally, while all possible transformations of the same type can be executed concurrently, different types of transformations must be executed in the predefined order. However, since

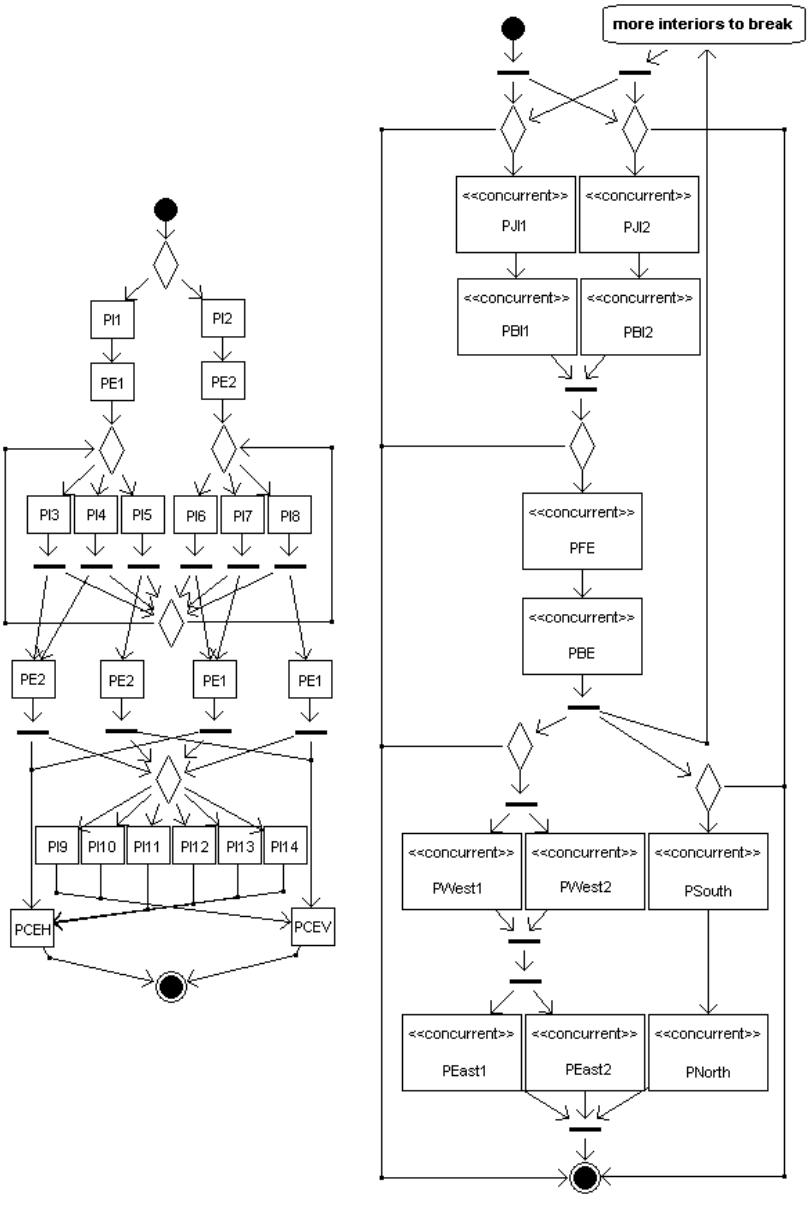


Fig. 5. Left panel: Control diagram based model of concurrency for the process of the generations of triangular finite elements two dimensional mesh. **Right panel:** Control diagram based model of concurrency for the process of h adaptation.

the mesh elements may have to be refined in portions (enforced by the mesh regularity rule), a new portion of refinements can be started while adjacency propagation of the previous portion is still running.

4 Petri Nets Model of Mesh Transformations

In this section we introduce the Petri net based model of concurrency for the initial mesh generation as well as for h adaptation. In addition to the control diagram based model it provides analysis tools typical for Petri nets like reachability graphs, place/transition invariants, etc. A system modeled as a Petri net can be analyzed for deadlocks, starvation or infinite execution. This model also allows to express some quantitative relationships between graph grammar productions.

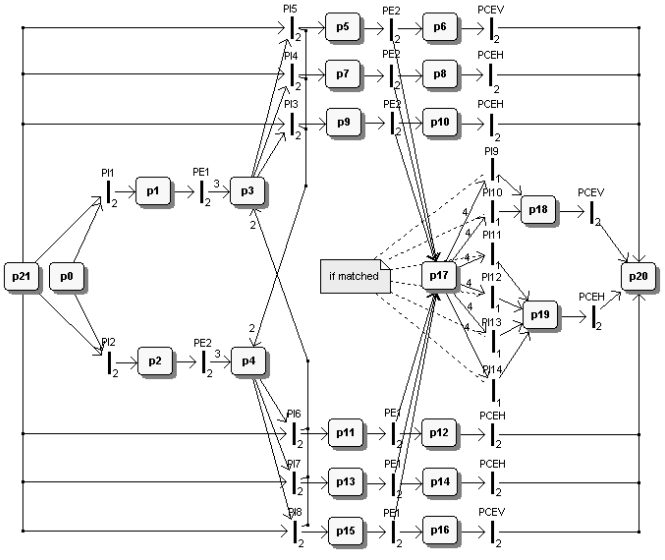


Fig. 6. Petri nets model of concurrency for the process of triangular finite element two dimensional mesh generation

Figure 6 presents a priority Petri net with guards for initial mesh generation. Transitions are named after productions they execute. Priorities are assigned to transitions (in subscript) to enforce certain sequence of execution. Transitions **PI9** - **PI14** (priority 1) should be executed as soon as they get activated, before any other transition being active at the same time. The prioritization has been introduced to ensure correct mesh construction (with no overlapping elements). Transitions **PI9** - **PI14** have an additional activation guard - they can be activated only if there is an actual matching for these productions in the mesh being generated. The condition expressed with the guard is not enforced by the net itself. Edge weights denote the number by which the marking of the destination place is increased or the marking of the origin place is decreased. The meaning of some of the places is as follows:

- *p3* number of type 2 elements that can be produced from current mesh shape
- *p4* number of type 1 elements that can be produced from current mesh shape
- *p21* number of elements left to be produced

The initial marking is: p0 - 1, p21 - N, where N is a parameter to the system telling how many elements the initial mesh will consist of. The rest of the places are empty. Just one token in place p0 implies that only one of the starting transitions (**PI1** or **PI2**) may be fired since they are in conflict. The only transitions that increase the number of tokens in the net are **PE1**, **PE2** and **PI3** - **PI8**. Only one of the transitions: **PE1** input to place p3 or **PE2** input to place p4 may be fired and it may be fired only once due to the starting transitions conflict. Transitions PI3 - PI8 have place p21 as input which limits the number of their activations since the initial marking of that place is only decreasing during the system execution. Therefore the Petri net is bounded and the initial mesh generation will eventually stop.

The reachability graph is finite and suitable for analysis. Place p21 is the determinant of the net's liveness. When that place gets empty, the net quickly reaches a dead marking - as soon as places p5, p16 and p18 also get empty. Such a dead marking means the end of initial mesh generation; no deadlock is possible while place p21 contains tokens. The Petri net structure is fixed; different shapes of generated meshes will result in choosing different paths in the reachability graph.

Figure 7 presents the Petri net for h adaptation. The initial marking of place p0 is the number of the mesh elements chosen for adaptation by an external

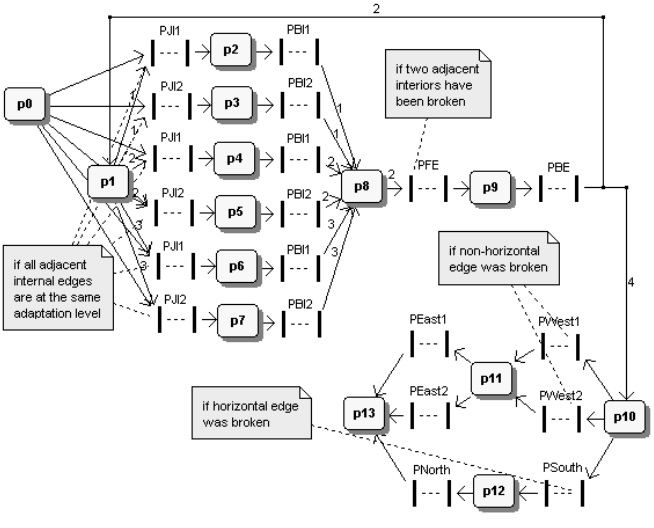


Fig. 7. Petri nets with guards model of concurrency for the process of h adaptation

driver. This includes elements that have to be broken as prerequisites to the adaptation of the chosen elements (according to the mesh regularity rule). The marking of place p1 determines the adaptation "potential" of the mesh and initially it will be: number of elements with 3 neighbors multiplied by 3 + number of elements with 2 neighbors times 2 + number of elements with 1 neighbor. The marking of places p1 and p8 should be preserved between subsequent iterations of h adaptation.

The branch with places p2 and p3 models interior breaking of elements with 1 neighbor; branch with places p4 and p5 - elements with 2 neighbors; branch with places p6 and p7 - elements with 3 neighbors. Two transitions linked with a dashed line denote actually all activated transitions of a given type fired concurrently. Although their numbers depend on the actual problem being solved, the reachability graph is still finite and can be analyzed since the net is bounded thanks to the place p0. Mesh regularity rules are enforced with the transition guards. Therefore the net can reach a dead state even if places p1 and/or p8 still contain tokens.

5 Conclusions

In this paper, we presented an extended graph grammar model, dedicated for the self-adaptive *hp*-FEM algorithm with an arbitrary initial regular triangular mesh. We analyzed both algorithms for an initial mesh generation and mesh adaptation. We created the model of concurrency, based on control diagrams defining the relations between graph grammar productions. The Petri nets were created, modeling both initial mesh generation and adaptation process. This will provide the analysis tools typical for Petri nets like reachability graphs, place/transition invariants, etc. The Petri nets model of the system can be analyzed for deadlocks, starvation or infinite execution.

The future work in this field will include the analysis of concurrency potential of the remaining steps of *hp*-FEM. It may also be beneficial to employ other concurrency modeling tools for expressiveness comparison.

Acknowledgments. The work reported in this paper was partially supported by Polish Ministry of Scientific Research and Information Technology, and by the Foundation for Polish science under Homming program.

References

1. Grabska, E.: Theoretical Concepts of Graphical Modeling. Part One: Realization of CP-Graphs. Machine Graphics and Vision 2(1), 3–38 (1993)
2. Grabska, E.: Theoretical Concepts of Graphical Modeling. Part Two: CP-Graph Grammars and Languages. Machine Graphics and Vision 2(2), 149–178 (1993)
3. Grabska, E., Hliniak, G.: Structural Aspects of CP-Graph Languages. Schedae Informaticae 5, 81–100 (1993)

4. Demkowicz, L.: Computing with hp-Adaptive Finite Elements. Chapman & Hall/Crc Applied Mathematics & Nonlinear Science, vol. I (2006)
5. Demkowicz, L., Kurtz, J., Pardo, D., Paszynski, M., Rachowicz, W., Zdunek, A.: Computing with hp-Adaptive Finite Elements. Chapman & Hall/Crc Applied Mathematics & Nonlinear Science, vol. II (2007)
6. Paszyński, M., Kurtz, J., Demkowicz, L.: Parallel Fully Automatic hp-Adaptive 2D Finite Element Package. Computer Methods in Applied Mechanics and Engineering 195(7-8)(25), 711–741 (2006)
7. Paszyński, M., Demkowicz, L.: Parallel Fully Automatic hp-Adaptive 3D Finite Element Package. Engineering with Computers 22(3-4), 255–276 (2006)
8. Paszyński, M., Paszyńska, A.: Graph transformations for modeling parallel *hp*-adaptive Finite Element Method. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Wasniewski, J. (eds.) PPAM 2007. LNCS, vol. 4967, pp. 1313–1322. Springer, Heidelberg (2008)
9. Paszyńska, A., Paszyński, M., Grabska, E.: Graph transformations for modeling *hp*-adaptive Finite Element Method with triangular elements. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 604–613. Springer, Heidelberg (2008)

Multi-agent Crisis Management in Transport Domain

Michał Konieczny, Jarosław Koźlak, and Małgorzata Żabińska

Department of Computer Science,
AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland
{kozlak,zabinska}@agh.edu.pl, michal.t.konieczny@gmail.com

Abstract. A multi-agent system that solves static and dynamic versions of transport problem (Pickup and Delivery Problem with Time Windows) in presence of crises is shown here. Scenarios to serve different kinds of crises (especially vehicle failures and traffic jams) by agents are described. The results summarising system functioning for solving classical PDPTW as well as influence of crises and applied algorithms of serving them upon quality of obtained solutions have been presented.

1 Introduction

Efficient transport of people and goods is an important issue with different transport companies. For some branches of the industry costs of transport may reach even 70 % of value of transported goods. Therefore it is justified to tend to minimise transport costs which influence the final price of goods and services.

Many transport problems may be treated as PDPTW, which is an extension of the widely examined Vehicle Routing Problem. Solving the PDPTW consists in finding routes which may be served by the minimal number of vehicles and to minimise the distance travelled. The distance is the total distance vehicles have to travel to serve the existing set of transport request between points of pick up and delivery, on the assumption that each request will be served within the time range defined by the time window, whereby the maximum capacity of vehicles is not exceeded. Since the problem has a large complexity (NP-problem), the research on solution of PDPTW concentrates rather on working out proper meta-heuristic methods than upon searching for exact solutions.

The purpose of the work carried out is to create the system which enables solving PDPTW with more accurate reflection of real conditions. PDPTW will be extended, among others, by a complex transport network, dynamics of transport requests as well as crises - unexpected events, which force changes of routes of transport teams. Objects, existing at the logistic firm, such as dispatching of tasks, vehicles, or being responsible for crisis situations service, may be intuitively implemented with the use of agents.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 855–864, 2009.

2 State of the Art

2.1 Transport Problems and Their Solving

A frequently used approach to solve complex computational problems is to divide the task into two phases. In the first one, a relatively good primary solution is generated, whereas in the second phase, it is optimised. the following are popular construction heuristics which build primary solutions: Insertion Heuristic, Sweep Heuristic and Partitioned Insertion Heuristic described in [8]. Methods which improve the quality of obtained solution are based on multiple modifications of a set of routes with the use of local changes consisting in moving requests from the route (SPI - Single Paired Insertion), exchange of requests between routes (SBR - Swapping Pairs Between Routes) and change of sequence of requests for a given route (WRI - Within Route Insertion). The optimisation process may be managed using algorithms like tabu search [9], simulated annealing, evolutionary algorithms, Squeaky Wheel Optimisation [10] and Ant Colony optimisation [5].

2.2 Multi-agent Solutions

Agent systems which solve transport problems are based on two schemes: Contract Net [12] and simulated trading. An algorithm of iterative optimisation (Simulated Trading) [3] is used to improve the primary feasible solution. It applies a stock exchange mechanism, where vehicles optimise their plans by subsequent purchase and selling of transport requests. MARS [6] is an multi–agent system realising transport planning. Its purpose is to organise the cooperation between some transport companies, so as to serve the set of transport tasks in an optimal way.

TeleTruck is a system modelling process at the transport company. It is a holonic multi-agent system created by the firm DFKI [4]. TeleTruck has a richer functionality than MARS. It is a prototype of an application built with the essential help of a transport company. It plans real requests with the use of heterogenic agents modelling different vehicle forms. The main goal of TeleTruck is to model basic objects of the real world (drivers, trucks, trailers, containers) with the use of basic agents. These agents have a task to merge themselves and create holonic agents, cooperating together and performing tasks.

2.3 Crises in Transport Systems

The specific features of multi-agent systems, connected with decision making based on local, partial knowledge and dynamic system reorganisation in response to changes taking place, favour this approach when reacting to crisis situations. Here, we will especially focus on applications in the transport domain. In [11] a general idea of the model of such a system to react on crisis situations, prevent them and minimise consequences is given. The crisis situations analysed are most often delays caused by uncertain travel times and traffic jams [7]. Furthermore, the presented MARS system was equipped with a module responsible for modelling and predicting traffic jams while constructing routes.

3 Concept of Multi-agent System

The main task of the constructed system is to solve static and dynamic transport problems (PDPTW) and additionally recognising, serving and minimising disadvantageous results of the chosen types of crises situations. In such a kind of problem it is possible to distinguish units with different degrees of autonomy of action (dispatcher, vehicles), which cooperate to realise some common goals. This is the reason why an approach using multi-agent systems is applied. The environment, where the agents act is either the Euclidean space or a graph representing road connections.

3.1 Agents' Description

In the constructed system the following main types of agents exist: Dispatcher Agent, Execution Unit Agent (EUnit Agent) and Crisis Manager Agent.

3.2 Dispatcher Agent

Dispatcher Agent AD plays a role of manager in the system. It controls the work and life cycle of Execution Unit agents, as well as it distributes transport requests between them in the best possible way. In the system only one Dispatcher Agent exists.

$$AD = (G_D, K_D, A_D) \tag{1}$$

where:

G_D - agent's goal which is to minimise the function nv+TD, where v – number of vehicles, TD –total distance, n – coefficient.

K_D - agent's knowledge, $K_D = \{Reqs, StatReqs, Env, EnvStat\}$, where $Reqs$ – information about requests, $StatReqs$ – information about status of requests, which assigns to each of them a value from the set {received, rejected, allocated, picked–up, delivered, wait_trainshipment}, Env — information about transport network, $EnvStat$ — information about the current status of transport network,

A_D – actions to be realised, $A_D = \{AquireEU, AllocateReq\}$

3.3 Execution Unit Agent

The agent AEU manages the transport unit which realises transport requests. Together with other units it participates in auctions of transport requests, and then it realises the assigned requests.

$$AEU = (G_{EU}, P_{EU}, Loc, S_{EU}, K_{EU}, A_{EU}) \tag{2}$$

where:

G_{EU} – agent's goal which is to realise the maximal possible number of requests as well as minimising of the total distance,
P_{EU} – planned travel route,
Loc – current agent's position,
S_{EU} – agent's status, information about allocated requests,
K_{EU} – agent's knowledge $K_{EU} = \{Reqs, Env, EnvStatg\}$ comprising $Reqs$ — set of requests, Env — information about environment (transport network), $EnvStat$ — information about status of environment taking into account information concerning traffic jams as well as parts of a network totally exluded from traffic,
A_{EU} – actions of $A_EU = \{ReqPart, PickUp, Delivery, SendInfo\}$, where $ReqPart$ — participation in auctions of transport requests, $PickUp$ — loading, $Delivery$ — unloading, $SendInfo$ — transfer of information about discovered crisis situations to Crisis Manager Agent.

3.4 Crisis Manager Agent

Crisis Manager Agent ACM is dedicated to manage crisis situations originating in the system. The agent is responsible for the detection of crisis situations as well as handling them. Detection of the crisis situation may be done directly by ACM, or indirectly by information from another agent. Handling of crisis situations consists in such collaboration with other agents, so as to minimise all negative results of the situation. Currently, only one Crisis Management Agent exists in the system.

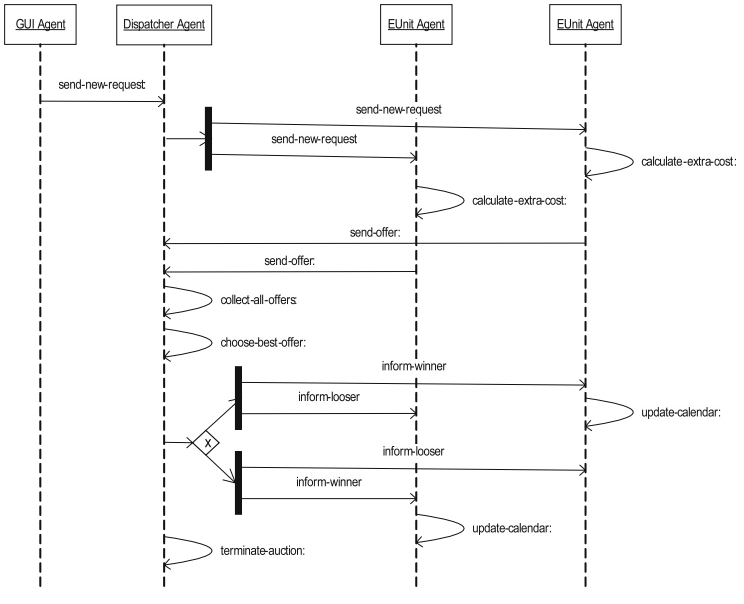


Fig. 1. Request allocation without crises handling

$$ACM = \{K_{ACM}, R_{ACM}, A_{ACM}\} \tag{3}$$

where:

K_{ACM} – knowledge about crisis situations
R_{ACM} – decision rules describing actions to be undertaken, for a given crisis situation and situations occurring during its service
A_{ACM} – actions performed by an agent, $A_{ACM} = \{SendInf, RecStat, RReal\}$, $SendInf$ — sending the information about crisis situation, $ResStat$ — receiving information about the status of the crisis situation, $RReal$ — proposal to reassign transport requests.

3.5 Request Allocation Protocol

In fig. 1, the request allocation to vehicles without crises handling is shown.

4 Crises Handling

Crisis situations are defined as any unpredictable events which have a significant impact on company activity. They include transport fleet faults, delays of receiving or delivering goods by customer, as well as unpredictable changes in road infrastructure, from intensive road traffic, which significantly increase the travel time, or faults of road sections. Occurrence of a crisis situation forces change of transport plan or forces the use of a larger amount of transport units.

4.1 Types of Crises

Crisis situations may be distinguished into three groups: crisis situations in transport requests, transport fleet and road infrastructure. The first group includes request withdrawal and delay in receiving commodity. The second group includes failures of the transport unit. The last group includes traffic jams and road blocks.

4.2 Transport Unit Failure

The discussed crisis situation involves temporary or permanent exclusion of a transport unit from use. Detection of this crisis situation involves Crisis Manager Agent "finding out" at first about the occurrence of this crisis situation. Next, Crisis Manager Agent informs transport unit about the situation occurred. In cases when the expected time of repair of the defective transport unit is short enough to let all planned requests be realised in given time windows, then the crisis situation does not significantly affect the course of simulation. It only extends the total time of realising the whole transport plan. However, if a failure of the transport fleet unit is serious, it must be replaced with an adequate vehicle, and transport requests in its transport plan must be passed on to other transport units. Requests may be transferred on to other transport

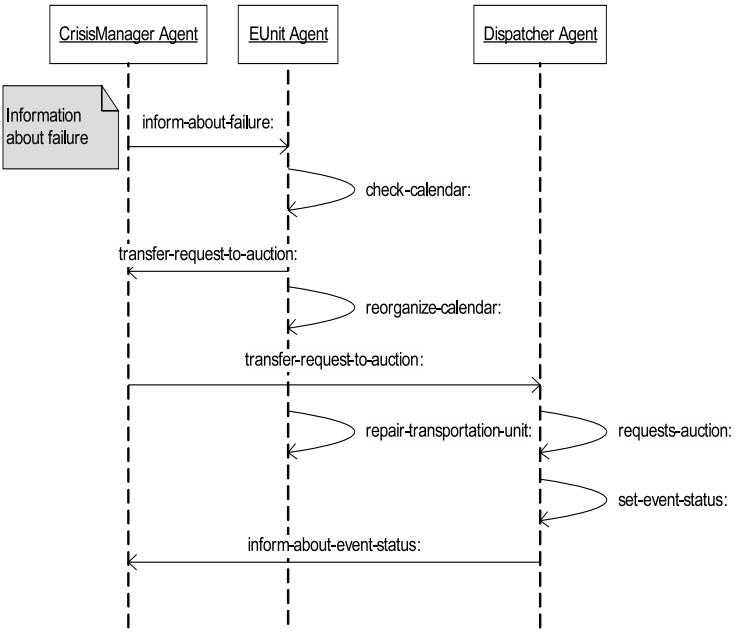


Fig. 2. Interactions: Handling of vehicle failure

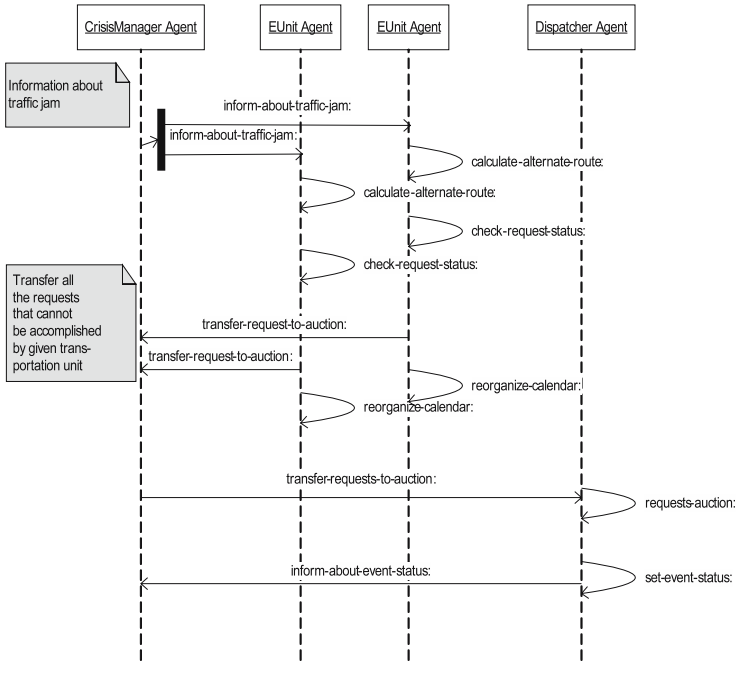


Fig. 3. Interactions: Handling of traffic jam

units. The faulty unit must be transported to a point of repair, which increases the cost of realising the transport plan. In cases when the transport set failure is serious (long time of repair), it is necessary to put the next transport requests in the transport plan up for auction, so that the time of realising them closes in a definite time window.

4.3 Traffic Jam

Traffic jams have significant influence on shipping agent activity. Long-lasting jams contribute to delays in realising the transport requests in given time windows. Detection of a crisis situation such as a traffic jam may be performed in two ways. (1) A transport set moving along a section of the transport network, on which a traffic jam occurs, informs Crisis Manager Agent about this event. (2) Crisis Manager Agent can receive information about this event directly from the crisis situations generator. After the traffic jam has been detected, there are three alternative scenarios presented in fig. 3, depending on the certain situation. (1) If the delay resulting from a traffic jam has no influence on realising the further transport plan by a transport unit, the unit should continue without any changes. (2) If the delay results in a subsequent request in the transport plan to be delayed, the transport plan should be changed, to avoid this kind of situation. The change of the plan involves determining alternative routes, bypassing the section with traffic jam (3) If, despite determining an alternative route, subsequent request realisations are delayed, they should be passed on to other transport units.

5 Realisation

The system architecture scheme is presented in fig. 4. JADE platform (Java Agent Development Framework) version 3.3 [2] has been used. Fig. 4 presents all

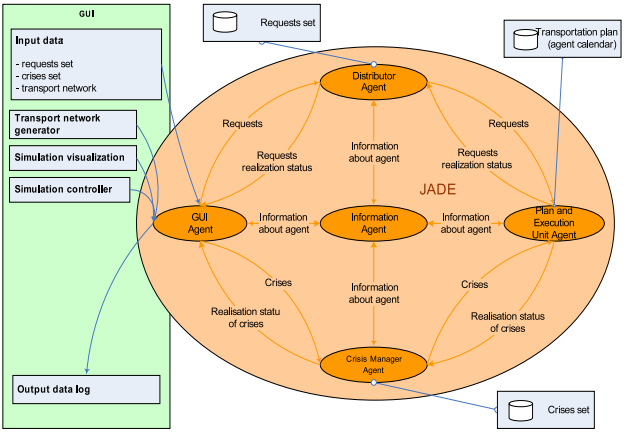


Fig. 4. System architecture

types of agents existing in the system and JADE platform range. It also presents data stored by certain agents during the simulation. In Dispatcher Agent's case, it constitutes a set of transport request to be realised, as well as information considering the status of their performance. Each agent of the Planning and Executive Units, has a transport plan. A set of crisis situations to serve together with their status, is stored by Crisis Manager Agent.

Transport network generator delivers environment for agents in the system.

6 Experiments and Results

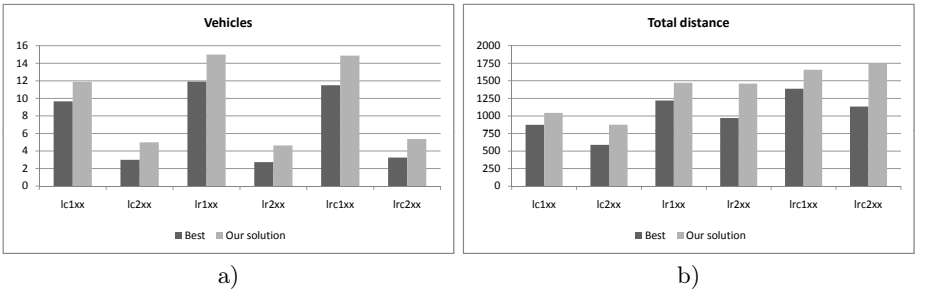
The input data for the system may be distinguished into three categories. The first concerns transport requests, the second – road networks, connected points of pickup or delivery and dispatch centre as well as connections between them, and the third concerns crisis situations.

Evaluation of the algorithms applied to solve PDPTW was done on the basis of benchmarks, which contain sets of transport requests [1], developed by Li and Lim [9]. The published sets of tests belong to the three kinds of problems: C – locations placed in clusters, R – locations placed randomly or RC – locations placed either in clusters or randomly. Additionally, each kind of problems may have small or large time windows and may have different numbers of requests to be served.

6.1 Solving of PDPTW Static Problems

In this section selected results of solving PDPTW for static and dynamic cases are described. In fig. 5 the average number of used vehicles and the average total distances for selected groups of benchmarks are presented. The results obtained by the created algorithm are not much worse than the best known solutions [1].

The following experiments lead in the direction of using the maps of transport roads generated according to prepared algorithms considering requests points from Li &Lim benchmarks and an analysis of the dynamic version of the problem.



a) b)

Fig. 5. Results: Static PDPTW: a)Vehicles needed b)Total travel distance

6.2 Crises

The results will be evaluated by the number of used vehicles and the total travel distance as well their increase in comparison to the experiments without crisis situations. In each of the performed tests 10 crises situations of each analysed kind have been randomly generated. After performing the series of tests, numbers of non-performed requests for the given kinds of crises situations are: 10 (19.4%) for request withdrawal, 2.1 (4.1 %) for the delay of the delivery/receipt of the package by customer, 13,7 (26.6%) for vehicle breakdown, 4.6 (8.9%) for the increase of the travel time caused by traffic jams and 5.2 (10.1%) for closed routes.

In fig. 6a the number of used vehicles, the average for each group of request, during the realisation of requests with different kinds of crisis situations introduced are shown. In the vertical axis the number of vehicles used is presented and in the additional vertical axis on the right the same value in comparison to the number of vehicles used during the realisation of requests without crisis situations is shown. In fig. 6b the average distances travelled by all vehicles, for each group of requests during the realisation of requests with different kinds of crisis situations introduced are presented. In the vertical axis the distance travelled by vehicles of the fleet is presented and on the right side of the figure, the same value in comparison to the distance travelled by the vehicles, during the realisation without crisis situations is shown.

The largest decrease of the number of performed requests and the increase of the used vehicles took place for the cases with vehicle failures and road closures. The results for problems with crises such as customer delay and increase of travel time are burdened by relatively low additional costs. The increase of travel times and closing of routes especially lengthened the total distance because it forced a search for other, longer routes.

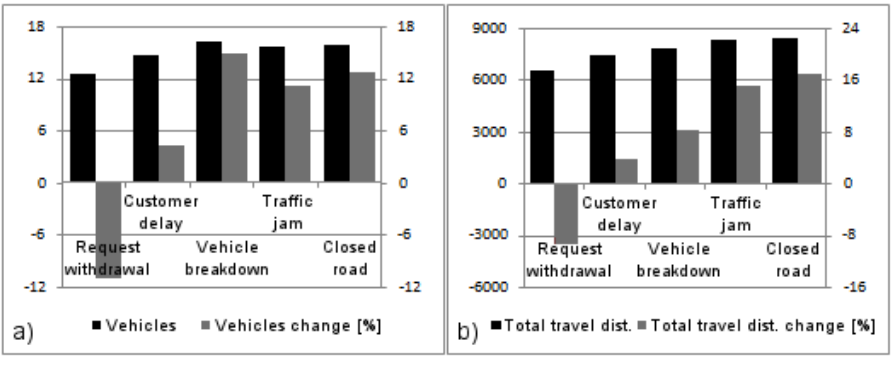


Fig. 6. Results for the different kinds of crises: a)Vehicles needed b)Total travel distance

7 Conclusions

During this work, a multi-agent system for solving transport problem PDPTW placed under the real conditions and with features of solving several kinds of potential crisis situations has been realised.

Future work will focus on the improvement of the quality of optimisation algorithms, introduction of soft time windows and the applications of holons. The soft time windows will make it possible to build plans which take into consideration fees dependent on the size of the delay. The application of holonic approach makes it possible to take into consideration the transport units composed of different components so as to be best adjusted to needs.

References

1. Benchmarks - Vehicle Routing and Travelling Salesperson Problems, http://www.sintef.no/static/am/opti/projects/top/
2. Java Agent DEvelopment Framework, http://jade.tilab.com/
3. Bachem, A., Hochstattler, W., Malich, M.: Simulated Trading A New Parallel Approach for Solving Vehicle Routing Problems. In: Proceedings of the International Conference Parallel Computing: Trends and Applications (1994)
4. Burckert, H.-J., Fischer, K., Vierke, G.: Transportation scheduling with holonic MAS - the TELETRUCK approach. In: Third International Conference on Practical Applications of Intelligent Agents and Multiagents, PAAM 1998 (1998)
5. Doerner, K., Hartl, R.F., Reimann, M.: Ant Colony Optimization applied to the pickup and delivery problem. Technical Report Working Paper 76, Departament of Managment Science, University of Vienna, Wien, Austria (November 2000)
6. Fischer, K., Muller, J., Pischel, M.: Cooperative Transportation Scheduling: an Application Domain for DAI. In: Applied Artificial Intelligence, pp. 1–33 (1996)
7. Koźlak, J.: Learning in cooperating agents environment as a method of solving transport problems and limiting the effects of crisis situations. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4488, pp. 872–879. Springer, Heidelberg (2007)
8. Lau, H., Liang, Z.: Pickup and Delivery with Time Windows: Algorithms and Test Case Generation. In: Proceeedings of 13th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2001 (2001)
9. Li, H., Lim, A.: A Metaheuristic for the Pickup and Delivery Problem with Time Windows. In: Proceedings of 13th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2001), Dallas, USA (2001)
10. Lim, H., Lim, A., Rodrigues, B.: Solving the Pick up and Delivery Problem using "Squeaky Wheel" Optimization with Local Search. In: Proceedings of American Conference on Information Systems, AMCIS 2002, USA (2002)
11. Nawarecki, E., Koźlak, J., Dobrowolski, G., Kisiel-Dorohinicki, M.: Discovery of crises via agent-based simulation of a transportation system. In: Pěchouček, M., Petta, P., Varga, L.Z. (eds.) CEEMAS 2005. LNCS, vol. 3690, pp. 132–141. Springer, Heidelberg (2005)
12. Smith, R.G.: The contract net protocol: high-level communication and control in a distributed problem solver. IEEE Transactions on Computer, 1104–1113 (December 1980)

Agent-Based Model and Computing Environment Facilitating the Development of Distributed Computational Intelligence Systems

Aleksander Byrski and Marek Kisiel-Dorohinicki

AGH University of Science and Technology, Kraków, Poland
{olekb,doroh}@agh.edu.pl

Abstract. In the paper a simple formalism is proposed to describe the hierarchy of multi-agent systems, which is particularly suitable for the design of a certain class of distributed computational intelligence systems. The mapping between the formalism and the existing computing environment *AgE* is also sketched out.

1 Introduction

Realization of multi-agent systems is usually performed utilizing in parallel: a dedicated formalism for describing its structure and behavior, and then general-purpose design and implementation methods, such as UML and object-oriented programming to achieve a functional system [1]. Of course using general-purpose design and implementation methods has an unquestionable drawback, which is the need to establish a mapping between the model and its implementation each time a new class of systems (or even a novel approach to modeling) is considered. Obviously dedicated implementation tools help to understand the designed system faster and allow to extend its features easier. There exist some frameworks, which facilitate the construction of agent systems, such as Jade, Aglets, or Grasshopper [2]. Other approaches concentrate on specific aspects of agent systems, like AgentSpeak language for describing BDI agent behavior together with its implementation based on Jason and Moise+ facilitate constructing agents interacting in some organizational structure [3,4]. However, it is obvious that such frameworks are not equally useful for implementing all kinds of multi-agent systems.

Our interests focus on a specific class of agent systems, which use computational intelligence paradigms — particularly hybrid techniques based on the concept of decentralized evolutionary computation [5]. Several variants of these techniques were considered for different goals and problems (e.g. global, multimodal and multi-criteria optimization), and a variety of systems were developed using general-purpose tools. Both the realized systems and proposed models proved useful for the analysis of these techniques — in testing their efficiency for different kinds of problems or exploring their asymptotic behavior [6]. However, still a complete methodology which would bring together the model and its implementation is lacking. Thus in this paper a simple formalism is proposed to describe the particular class of computing multi-agent systems, which is suitable

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 865–874, 2009.

for their realization. It is aimed to describe the structure and behavior of the system, yet it does not provide the means for its more sophisticated analysis. A mapping is established between the proposed formalism and an existing computing environment, which was successfully used in the implementation of the above mentioned decentralized evolutionary computation systems.

The paper begins with the presentation of a formalism which allows for describing the structure and behavior of computing agent-based systems. The model of an evolutionary multi-agent system is given as an illustration in the next section. Finally the mapping between the model and the computing environment *AgE* is shown and some conclusions are drawn.

2 Structure and Behavior of a Computing MAS

Obviously a multi-agent system consists of autonomous agents. Thus the proposed model defines MAS as a set of agents, but also a set of actions to be executed by the agents, and the environment represented by some common data, which may acquired by the agents:

$$AS \ni as = \langle Ag, Act, qr_1, \ldots, qr_m \rangle \tag{1}$$

where:

$Ag \subset AG$ is the set of agents of as,
$Act \subset ACT$ describes actions that may be performed by the agents of as,
$qr_i \in QR_i, i = 1, \ldots, m$ denote queries providing data (knowledge) available for all agents in the environment.

An agent ag may be described as the following tuple:

$$AG \ni ag = \langle id, tp, dat_1, \ldots, dat_n \rangle \tag{2}$$

where:

$id \in ID$ is a unique identifier of an agent[1],
$tp \in TP$ denotes the type of an agent, depending on its type, an agent is equipped with specific data and may perform specific actions,
$dat_i \in DAT_i, i = 1, \ldots, n$ represents problem-dependent data (knowledge) gathered by an agent (solutions, resources, outcomes of the observations etc.)

An agent may provide an environment for a group of other agents, which by themselves constitute a multi-agent system, which is essentially different then the one of the "parent" agent. These nested (multi-agent) subsystems introduce a tree-like structure, which will be further referred as *physical hierarchy* of agents. In fact, the agents of different subsystems form a structure, which is rather a

[1] For each element of the model its domain, which is a finite set of possible values, is denoted by the same symbolic name in upper case, e.g. ID is the set of all possible agent identifiers.

collection of trees (a forest), which may be perceived as a single tree with a virtual root node. Inner nodes of this structure may be perceived as *aggregates* of other agents, but of course it is only a simplification, since these sub-agents constitute a complete multi-agent system according to (1). Particular types of agents are often present at particular levels of physical hierarchy – the structure of allowed agent types in the physical hierarchy is called a *logical hierarchy*.

To identify an agent which provides the environment for a particular multi-agent system a mapping is introduced:

$$\gamma : AS \rightarrow AG \cup \{\varnothing\} \tag{3}$$

If a certain agent $ag \in Ag$ in $as = \langle Ag, Act \rangle$ provides the environment for $as^* = \langle Ag^*, Act^* \rangle$ the mapping will give $\gamma(as^*) = ag$, and this agent will be called an *aggregate* for as^*. For a multi-agent system at the top of the physical hierarchy $\gamma(as) = \varnothing$, since there is no real agent providing an environment for this system (symbol $\varnothing$ may be perceived as denoting a virtual root node of the hierarchy).

Let us introduce subsumption relation $\preceq$ in the set of types TP, which is used to state whether one type is the specialization of another:

$$"\preceq" \ \subset \ TP \times TP \tag{4}$$

Relation $\preceq$ defines a partial order in TP (it is reflexive, transitive, antisymmetric). Since the type of an agent determines its features, it is assumed that an agent with a more specific type is able to provide all data and perform all actions of an agent with a more general type, yet of course the realization of the descendant may be different than the parent.

Further deliberations are impossible without giving even the approximation of the system state space:

$$X = 2^{AS} \tag{5}$$

$$AS = 2^{AG} \times 2^{ACT} \tag{6}$$

Actual state space will be constrained by the existence of agent types, and the logical hierarchy of particular system, yet the precise description of the state of the whole system is beyond the scope of this paper.

Agents may perform actions in order to change the state of the system. An action is defined as the following tuple:

$$ACT \ni act = \langle tp, pre, post \rangle \tag{7}$$

where:

$tp \in TP$ denotes the type of agents allowed to execute the action (only agents of the type tp and descendant types – according to "$\preceq$" relation – may perform the action),

$pre \in X$ is the state of the system which allows for performing action act,

$post \in X \times X$ the relation between the state of the system before and after performing action act,

This tuple may be perceived as so-called Hoare's triple (precondition, operation, postcondition) used e.g. for contract definition in component-based software (see [7]). The descendant types (according to $\preceq$ relation) must hold the preconditions and postconditions of the parents, yet they may extend these conditions by adding new alternatives in preconditions and conjunctions in postconditions. When mentioned conditions are met, the descendant types are fully substitutable for parent types (in the means of contract substitutability [8]).

The following function family is used by an agent of type tp to choose the action to be executed:

$$\{\omega^{tp} : X \to \mathcal{M}(ACT)\} \tag{8}$$

The choice of the action is thus based on the current state of the system and is defined in stochastic terms[2]. Usually only small part of the system state will be taken into consideration (e.g. the agent's data or it's aggregate's data). Function ω allows an agent to choose only the actions with compatible (the same or descendant) type:

$$\forall ag \in AG : ag = \langle id, tp_1, dat_1, \ldots, dat_n \rangle, \forall act \in ACT : act = \langle tp_2, pre, post \rangle$$
$$\omega^{tp}(x)(act) > 0 \iff tp_1 \preceq tp_2 \tag{9}$$

thus introducing a new type of agents requires adding new function for choosing the actions.

To recapitulate, the relations among agents in the whole system may be described using two distinct hierarchies: physical hierarchy (aggregate agents provide an environment for nested multi-agent (sub)systems) and type hierarchy (each agent is assigned a type, which may be a specialization of another type).

3 Modeling Evolutionary Multi-Agent Systems

The idea of EMAS (Evolutionary Multi-Agent System) was proposed as a particular technique of decentralized evolutionary computation [9,5]. A variety of applications of this paradigm were considered from typical optimization problems to hybrid computational intelligence systems [10].

The system consists of individual agents decomposed into several subpopulations (demes). Agents contain (partial) solutions of the given optimization problem. They also contain a non-renewable resource called *life energy*, which is the base of a distributed selection process. Agents exchange their energy based on the fitness value of their solutions. These which gather more energy have greater chances for reproducing, and those with low energy have greater chances of dying. This energy-based selection is used instead of classical global selection mechanisms, because of the assumed autonomy of agents. Agents may also migrate to another subpopulation if they have enough energy.

To facilitate the realization of EMAS, two types of agents are used:

$$TP = \{ind, isl\} \tag{10}$$

[2] $\mathcal{M}(A)$ denotes the space of probabilistic measures over the measurable set A.

where *ind* denotes the type of an individual agent (as described above), and *isl* — of an aggregate agent, which introduced to manage subpopulations of individual agents (plays a role of an evolutionary island).

Consequently at the top of the physical structure of EMAS there is a system of evolutionary islands:

$$as = \langle Ag, \varnothing \rangle \quad \gamma(as) = \varnothing \tag{11}$$

where:

$$Ag \ni ag = \langle id, isl, Nb \rangle \tag{12}$$

and $Nb \subset Ag$ is the set of neighboring evolutionary islands, which is used to define the topology of migration.

Every evolutionary island *ag* provides an environment for the population of individual agents:

$$\forall ag \in Ag \;\; \exists as^* = \langle Ag^*, \{migr, get, repr, die\}, findAg, findLoc \rangle : \\ ag = \gamma(as^*) \tag{13}$$

and an individual agent is defined as:

$$Ag^* \ni ag^* = \langle id, ind, sol, en \rangle \tag{14}$$

where:

$sol \in SOL$ is the solution of the problem (usually for optimization problems $SOL \subset \mathbb{R}^n, n \in \mathbb{N}$),
$en \in \mathbb{R}^+$ is the amount of energy gathered by the individual,
$findAg : AG \to \mathcal{M}(AG)$ is the query that allows to choose the neighboring individual agent (another agent present in the same system),
$findLoc : AG \to \mathcal{M}(AG)$ is the query that allows to choose the neighboring island (using $Nb \subset Ag$).

The set of actions available for individual agents contains:

- *migr* – migration of an agent from one to another subpopulation,
- *get* – transfer of a portion of energy from one to another agent,
- *repr* – creation of a new agent by two parents,
- *die* – removing of an agent from the system.

Agents use following function to choose the action they intend to perform:

$$\omega^{ind} : X \to \mathcal{U}(\{migr, get, repr, die\}) \tag{15}$$

where $\mathcal{U}(\cdot)$ stands for a uniform random distribution.

Action of energy transfer *get* performed by $ag_1^* = \langle id_1, ind, sol_1, en_1 \rangle \in Ag^*$, which meets $ag_2^* = \underline{findAg}(ag_1^*) = \langle id_2, ind, sol_2, en_2 \rangle \in Ag^*$ (randomly chosen with uniform distribution from its neighbors) may be defined in the following way:

$$Act \ni get = \langle ind, pre, post \rangle \tag{16}$$

where:

$$pre = \#Ag^* > 1$$

$$post = (\varphi(sol_1) < \varphi(sol_2) \Rightarrow en_1' = en_1 + e_{get} \wedge\ en_2' = en_2 - e_{get}) \\ \vee\ (\varphi(sol_1) > \varphi(sol_2) \Rightarrow en_1' = en_1 - e_{get} \wedge\ en_2' = en_2 + e_{get})$$

where: $\varphi : SOL \to \mathbb{R}$ is the fitness function used for the evaluation of solutions, the precise definition of this function is problem-dependent. en_1' denotes the value of en_1 after performing the action[3], $\underline{A}(B)$ denotes the result of the random choice of the value from the set B with distribution A.

This action may be performed by an individual agent, which has at least one neighbor (agent in the same MAS subsystem). Depending on the quality of its solution, a part of energy is transferred from the worse to the better agent.

Action of reproduction *repr* performed by $ag_1^* = \langle id_1, ind, sol_1, en_1\rangle \in Ag^*$, which meets $ag_2^* = \langle id_2, ind, sol_2, en_2\rangle \in Ag^*$ (randomly chosen with uniform distribution from its neighbors) and produces the offspring agent $ag_3^* = \langle id_3, ind, sol_3, en_3\rangle \in Ag^*$ may be defined in the following way:

$$Act \ni repr = \langle ind, pre, post\rangle \tag{17}$$

where:

$$pre = \#Ag^* > 1$$

$$post = en_1 > e_{repr} \wedge\ en_2 > e_{repr} \Rightarrow Ag^{*\prime} = Ag^* \cup \{ag_3^*\} \\ \wedge\ en_3 = e_0 \wedge\ en_1{}' = en_1 - \frac{1}{2}e_0 \wedge\ en_2{}' = en_2 - \frac{1}{2}e_0 \wedge sol_3 = \underline{\chi}(sol_1, sol_2)$$

where: $\chi : SOL \times SOL \to \mathcal{M}(SOL)$ is a variation operator (recombination and mutation), its precise definition is problem-dependent.

This action may be performed by an individual agent with at least one neighbor, both with sufficient energy. They create a new agent based on their solutions.

Action of death *die* performed by $ag^* = \langle id, ind, sol, en\rangle \in Ag^*$ may be defined in the following way:

$$Act \ni die = \langle ind, pre, post\rangle \tag{18}$$

where:

$$pre = (en = 0)$$

$$post = (Ag^{*\prime} = Ag^* \setminus \{ag^*\})$$

This action may be performed by an individual agent which energy level falls to zero. This agent is removed from the system.

Action of migration *migr* performed by $ag^* = \langle id, ind, sol, en\rangle \in Ag_1^* : \exists\ as_1^* = \langle Ag_1^*, Act_1^*\rangle\ \wedge\ \gamma(as_1^*) = ag_1$ which migrates from $ag_1 = \langle id_1, isl, Nb_1\rangle$

[3] Here and later the symbol a' will denote the value of a after performing the action (usually a is used in precondition and a' only in postcondition). This is similar to *old* operator used in contract specification in Eiffel programming language [8].

to $ag_2 = \underline{findLoc}(ag_1) = \langle id_2, isl, Nb_2 \rangle : \exists\ as_2^* = \langle Ag_2^*, Act_2^* \rangle\ \wedge\ \gamma(as_2^*) = ag_2$ (ag_2 is randomly chosen from all neighbors of ag_1 with uniform distribution).

$$Act \ni migr = \langle ind, pre, post \rangle \tag{19}$$

where:

$$pre = ag_2 \in Nb_1$$

$$post = en_1 > e_{migr} \Rightarrow Ag_1^{*\prime} = Ag_1^* \setminus \{ag_1^*\} \wedge Ag_2^{*\prime} = Ag_2^* \cup \{ag_1^*\}$$

This action may be performed by an individual agent, located in the system with at least two evolutionary islands. The agent with sufficient energy e_{migr} leaves one island and moves to another.

4 AgE Computing Environment

The model presented constitutes a base for the design of the core structure of distributed computing environment AgE[4], developed as an open-source project at Intelligent Information Systems Group of AGH University of Science and Technology. The name reminds that it was primarily dedicated to agent-based evolutionary techniques (AgE = Agent Evolution), but now it grew up into a general-purpose platform facilitating the implementation of a variety of (not necessarily but most suitably) agent-based simulation and computing systems. Mainstream implementation is realized in Java and follows component approach, which allows for flexible (re)configuration of the system to meet the requirements of particular problems and solving techniques.

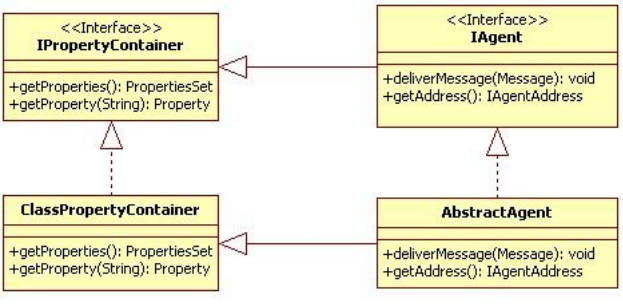


Fig. 1. Properties of objects in AgE computing core

Essential implementation entities that form the computing core of AgE (mainly agents) are realized so as they may be described in terms of *properties*. A property is a feature of an object, which may be referenced at runtime by its name

[4] http://age.iisg.agh.edu.pl/

— to be read, written or even monitored. This functionality is represented by `IPropertyContainer` interface and may be easily achieved extending an abstract class `ClassPropertyContainer` (Fig. 1), which provides instrumentation allowing to treat appropriately annotated methods (serving as getters or setters) or fields of a class as its properties.

According to the definition of an agent (2) properties allow for flexible access to agents' data dat_i, which facilitates the realization of observation of single agents. It is always possible because a fundamental interface representing an agent `IAgent` extends `IPropertyContainer`. Basic agent implementation `AbstractAgent` extends `ClassPropertyContainer` to gain this functionality without any additional effort. Also the identification of an agent, which is realized in the form of a unique agent address (`AgentAddress` class), is provided as a property.

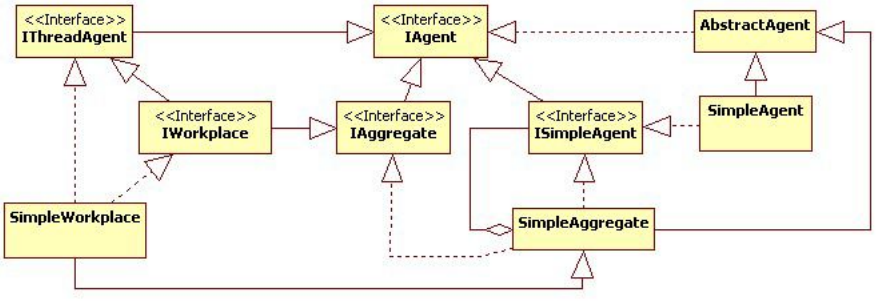


Fig. 2. Different kinds of agent implementations in AgE computing core

Fig. 2 shows different kinds of agent implementations – most notably those, which allow for parallel execution (`IThreadAgent`), as well as those which work semi-parallelly based on the concept of event-driven simulation (`ISimpleAgent`) and thus allow for more efficient realization of agent interactions. A specific interface (`IAggregate`) implies an agent which serves as an environment for the nested multi-agent subsystem. Since `IAggregate` extends `IAgent`, the implementation of a tree-like structure of agents introduced in (1) simply follows *composite* pattern.

The environment for descendant agents is available via dedicated interface `IAgentEnvironment` (see Fig. 3), which provides identification and both synchronous (queries) and asynchronous (messages) communication facilities. Queries may concern the state of the system an agent is a part of (realized by `IQueryable` interface) and the state of the "parent" agent's system (`queryParent()` operation). In case of event-driven execution (as implemented by `SimpleAgent` class) the environment (`ISimpleAgentEnvironment`) also provides the mechanism supporting actions (as realized by `SimpleAggregate` class).

Agents of the top-level multi-agent system (considering the described hierarchy nested subsystems) play special role in the organization of the platform — they work in parallel and may be distributed over the network. They may

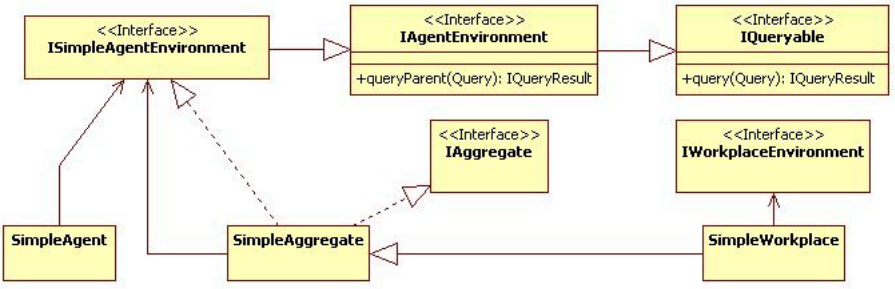


Fig. 3. Environments available in AgE computing core

be understood as roots of *local computing environments*, because all the agents in one branch of the hierarchy must be executed on one node (a single JVM in Java implementation) — thus they are marked by `IWorkplace` interface extending `IThreadAgent` and `IAggregate`. Obviously there is no special agent providing their environment, but rather there is some infrastructure available via `IWorkplaceEnvironment`, which provides identification and communication in the possibly distributed way.

5 Conclusions

In the paper a simple formal model of a computing multi-agent system was proposed and used to describe a specific computational intelligence system — an evolutionary multi-agent system. The proposed formalism is solely used for design, and that is why such details as the precise definition of the system state space or state transition functions were not taken into consideration. Since several existing EMAS implementations are based on AgE framework, the mapping between the proposed model and this framework was also shown in the course of the paper.

Further research should allow to extend the model to cover other computation intelligence techniques based on agent paradigm, such as iEMAS (immunological evolutionary multi-agent system) [11] or HGS (hierarchical genetic search) [12]. Also the mapping between the proposed formalism and existing models describing these techniques will be provided. Because the implementation of AgE framework continues, the formalism will surely be updated in the near future.

References

1. Booch, G., Rumbaugh, J., Jacobson, I.: The Unified Modeling Language User Guide. Addison-Wesley, Reading (1998)
2. Trillo, R., Ilarri, S., Mena, E.: Comparison and performance evaluation of mobile agent platforms. In: ICAS 2007: Proceedings of the Third International Conference on Autonomic and Autonomous Systems, Washington, DC, USA. IEEE Computer Society, Los Alamitos (2007)

3. Bordini, R., Hübner, J., Wooldridge, M.: Programming Multi-Agent Systems in AgentSpeak Using Jason. John Wiley & Sons, Ltd., Chichester (2007)
4. Hübner, J.F., Sichman, J.S., Boissier, O.: Developing organised multiagent systems using the moise+ model: programming issues at the system and agent levels. Int. J. Agent-Oriented Softw. Eng. 1(3/4), 370–395 (2007)
5. Kisiel-Dorohinicki, M.: Agent-oriented model of simulated evolution. In: Grosky, W.I., Plasil, F. (eds.) SOFSEM 2002. LNCS, vol. 2540, pp. 253–261. Springer, Heidelberg (2002)
6. Byrski, A., Schaefer, R.: Immunological mechanism for asynchronous evolutionary computation boosting. In: ICMAM 2008: European workshop on Intelligent Computational Methods and Applied Mathematics: an international forum for researches, teachers and students, Cracow, Poland (2008)
7. Szyperski, C.: Component Software: Beyond Object-Oriented Programming. Addison-Wesley Longman Publishing Co., Inc., Boston (2002)
8. Meyer, B.: Object-Oriented Software Construction. Prentice Hall PTR, Englewood Cliffs (2000)
9. Cetnarowicz, K., Kisiel-Dorohinicki, M., Nawarecki, E.: The application of evolution process in multi-agent world (MAW) to the prediction system. In: Tokoro, M. (ed.) Proc. of the 2nd Int. Conf. on Multi-Agent Systems (ICMAS 1996). AAAI Press, Menlo Park (1996)
10. Byrski, A., Kisiel-Dorohinicki, M., Nawarecki, E.: Agent-based evolution of neural network architecture. In: Hamza, M. (ed.) Proc. of the IASTED Int. Symp.: Applied Informatics. IASTED/ACTA Press (2002)
11. Byrski, A., Kisiel-Dorohinicki, M.: Immunological selection mechanism in agent-based evolutionary computation. In: Klopotek, M.A., Wierzchon, S.T., Trojanowski, K. (eds.) Intelligent Information Processing and Web Mining: proceedings of the international IIS: IIPWM 2005 conference, Gdansk, Poland. Advances in Soft Computing, pp. 411–415. Springer, Heidelberg (2005)
12. Schaefer, R., Kołodziej, J.: Genetic search reinforced by the population hierarchy. Foundations of Genetic Algorithms 7 (2003)

Graph Transformations for Modeling *hp*-Adaptive Finite Element Method with Mixed Triangular and Rectangular Elements

Anna Paszyńska[1], Maciej Paszyński[2], and Ewa Grabska[1]

[1]Faculty of Physics, Astronomy and Applied Computer Science,
Jagiellonian University, ul. Reymonta 4, 30-059 Cracow Poland
anna.paszynska@uj.edu.pl
[2]Department of Computer Science,
AGH University of Science and Technology,
Al. Mickiewicza 30, 30-059 Cracow, Poland

Abstract. The paper presents composition graph (CP-graph) grammar, which consists of a set of CP-graph transformations, suitable for modeling transformations of two dimensional meshes with rectangular elements mixed with triangular elements. The mixed meshes are utilized by the self-adaptive *hp* Finite Element Method (FEM) extended to support triangular and rectangular elements. The *hp*-FEM generates a sequence of mixed triangular and rectangular element meshes providing exponential convergence of the numerical error with respect to the mesh size. This is done be executing several h or p refinements over an initial mesh. The mixed finite element mesh is represented by attributed CP-graph. The proposed graph transformations model the initial mesh generation as well as mesh refinements. The proposed extended graph grammar has been defined and verified by using implemented software.

1 Introduction

The topological structure of the finite element mesh, with a hierarchy of vertices, edges and faces has been proposed by [1] to support mesh generation and data storage. The first attempt to model mesh transformations by applying the graph grammar concept has been proposed by [2] for the regular triangular two dimensional meshes with h adaptation. This has been done by using the quasi context sensitive graph grammar. However, the applicability of the quasi context sensitive graph grammar seems to be limited, since the mesh transformations utilized by adaptive algorithms are context dependable. The Composite Programmable graph grammar (CP-graph grammar) has been introduced by [3,4,5] as a tool for a formal description of various design processes. The CP-graph grammar expresses a design process by means of the graph transformations (called productions) executed on the CP-graph representation of designed objects. In this paper the CP-graph grammar is used to model mesh transformations executed by the 2D self-adaptive *hp* Finite Element Method (FEM) with triangular and rectangular finite elements [8]. The paper is an extension

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 875–884, 2009.

of the CP-graph grammar model introduced in [6] for rectangular finite element meshes, and the CP-graph grammar model introduced in [7] for triangular finite element meshes. The mixed triangular and rectangular element meshes are represented by attributed CP-graphs. The graph grammar consists of a set of graph transformations, called *productions*. Each production replaces a sub-graph defined on its left-hand-side into a new sub-graph defined on its right-hand-side. The left-hand-side and right-hand-side sub-graphs have the same number of free in/out bounds, and the corresponding bounds are denoted by the same index.

2 Graph Transformations

In the first part of this chapter, the subset of graph transformations, modeling generation of an arbitrary *hp* refined mesh, based on initial mesh with horizontal sequence of triangular and rectangular finite elements, is presented. The described graph transformations can be generalized into a case of arbitrary two dimensional initial mesh. The process of the initial mesh generation is expressed by the graph transformations **P1**, **P2**, **P3** and **P29** presented in Fig. 1. The graph vertices with **I2B** and **IQ** labels represent rectangular finite elements, while the graph vertices with **IT1** and **IT2** labels represent triangular elements. Each triangular finite element consists in three vertices, three edge nodes and one interior node. Each rectangular finite element consists in four vertices, four edge nodes and one interior node. The graph grammar productions presented in Fig. 1 generates the topology of the mesh. This is followed by the generation of the structure of finite elements, which is expressed by graph grammar productions **P7** and **P8** presented in Figures 2 and 3 for triangular elements and by graph grammar production **P31** presented in Fig. 4 for rectangular elements.

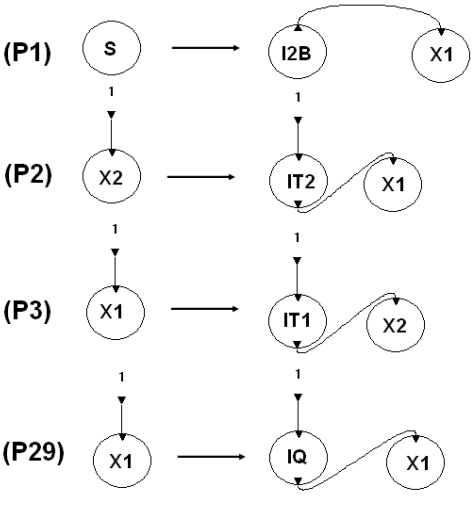


Fig. 1. Graph grammar productions responsible for an initial mesh generation

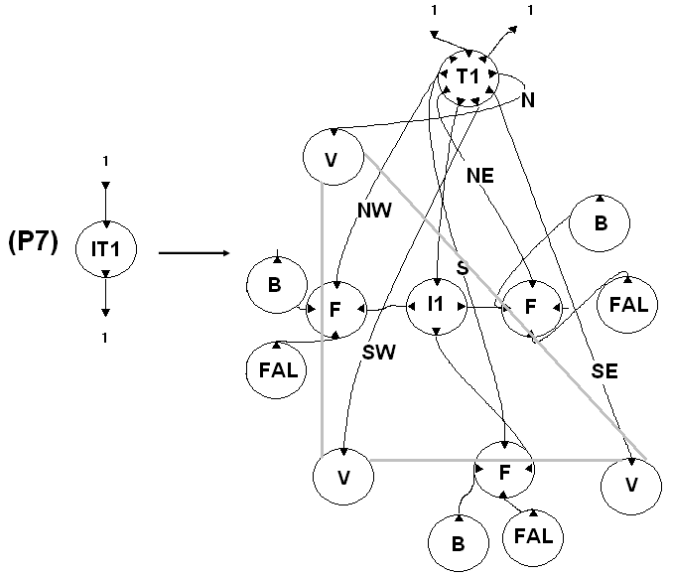


Fig. 2. Productions generating the structure of a triangular element of the first type

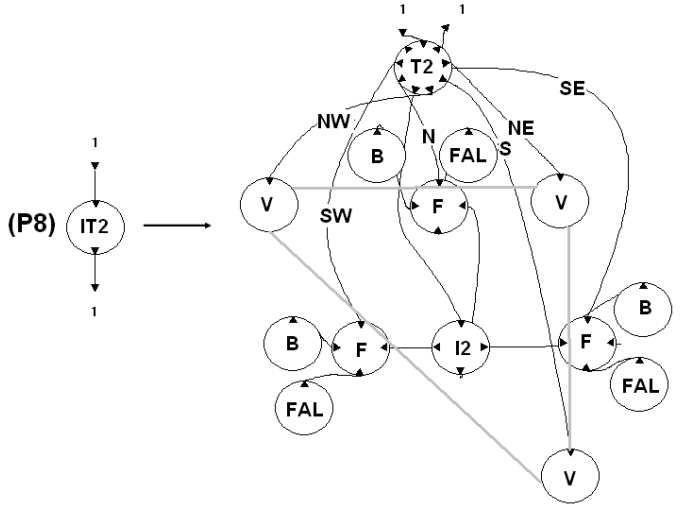


Fig. 3. Productions generating the structure of a triangular element of the second type

The **V** label stands for an element vertex, **F** stands for an element edge (face), **I**, **I1** and **I2** stand for interiors for three types of elements, respectively. If an element is adjacent to mesh boundary, then its free bounds are connected to **B** labeled graph vertex, denoting the boundary, as well as to **F** labeled vertex, denoting missed second father of the edge (edges located inside the domain have two father elements). We can identify common edges of adjacent finite elements.

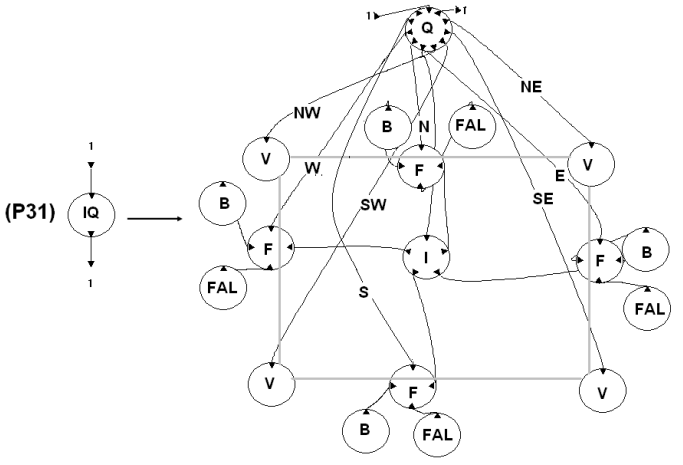


Fig. 4. Productions generating the structure of a rectangular initial mesh element

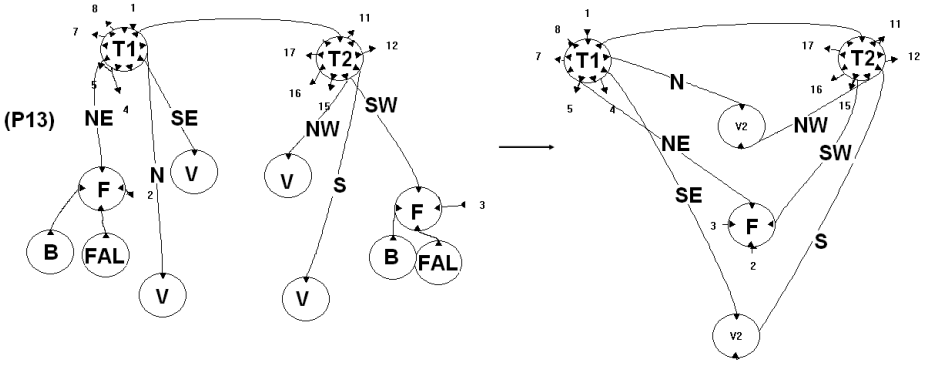


Fig. 5. Production for identification of the common edge of two adjacent triangular elements

The production **(P13)** that is identifying two adjacent triangular elements, and actually removing one duplicated edge, is presented in Fig. 5. The analogous production is utilize for identification of common edges for rectangular elements. The production **(P32)** that is identifying two adjacent elements, one rectangular and one triangular, and actually removing one duplicated edge, is presented in Fig. 6.

Once the structure of the mixed triangular and rectangular finite elements is generated, we can proceed with mesh refinements in the areas with strong singularities, where the numerical error is large. The decision about required refinements are made by the algorithm described in details in [8,9].

The refinement procedure is expressed by breaking element edges and interiors. To break a triangular element interior means to generate 4 new element interiors, and 3 new edges. To break a rectangular element interior means to

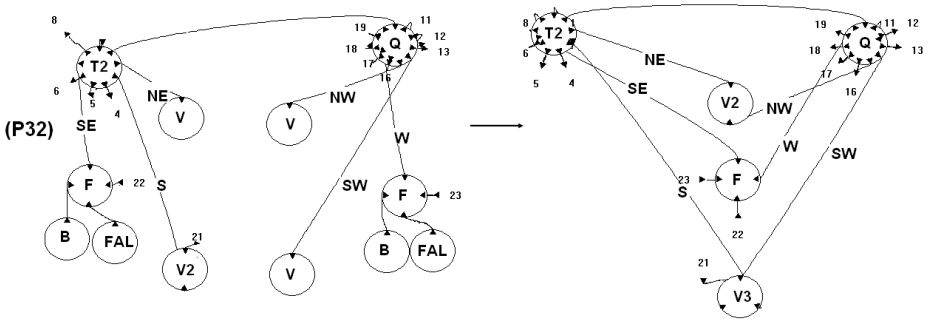


Fig. 6. Production for identification of the common edge of two adjacent elements, one rectangular and one triangular of the first type

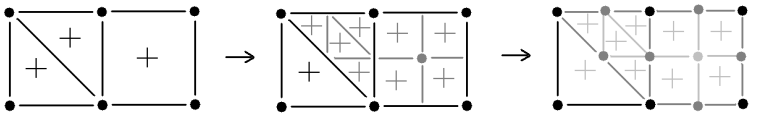


Fig. 7. The *h* refinement of selected triangular and rectangular elements: breaking of element interiors followed by breaking of element edges

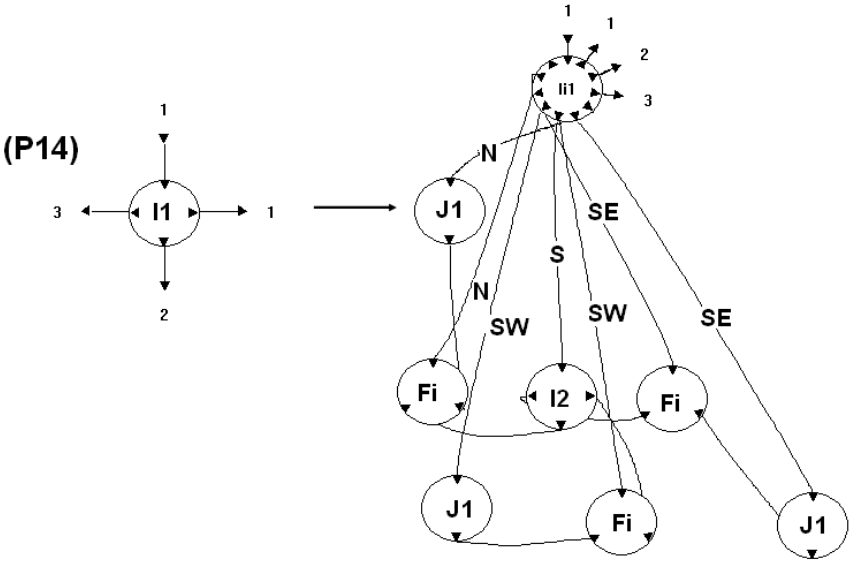


Fig. 8. Production breaking interior of first type triangular element

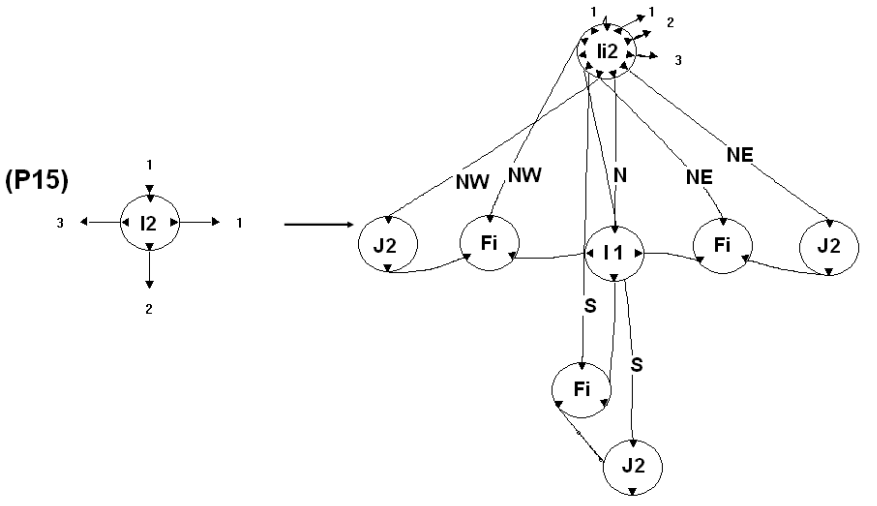


Fig. 9. Production breaking interior of second type triangular element

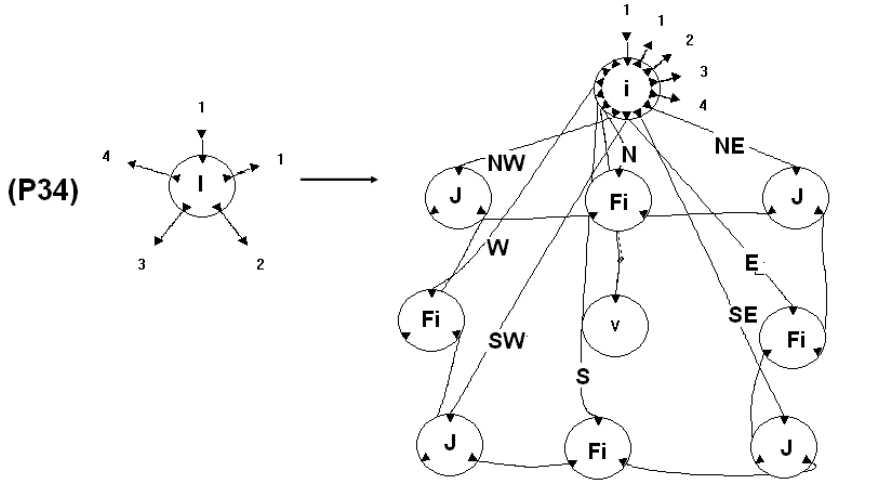


Fig. 10. Production for breaking rectangular element interior

generate 4 new element interiors, 4 new edges, and 1 new vertex. To break an element edge means to generate 2 new edge nodes and 1 new vertex. The procedure is illustrated in Fig. 7. The procedure of breaking of an element interior is expressed by **(P14)**, **(P15)**, **(P34)** productions presented in Figures 8-10. The newly created finite elements are represented by leaf nodes at the bottom level of generated refinement trees. The following mesh regularity rules are enforced during the process of mesh transformation, see [8]. An element edge can be broken only if two adjacent interiors have been already broken, or the edge

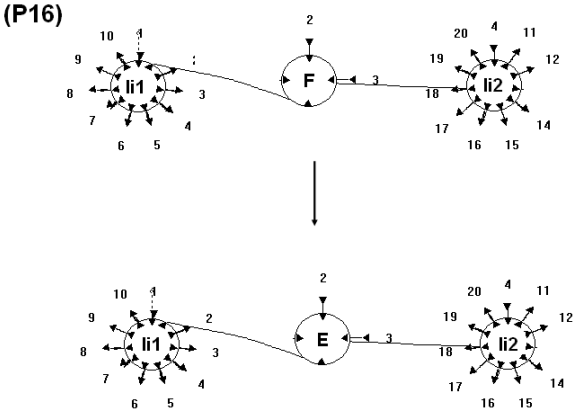


Fig. 11. Production allowing for breaking an element edge

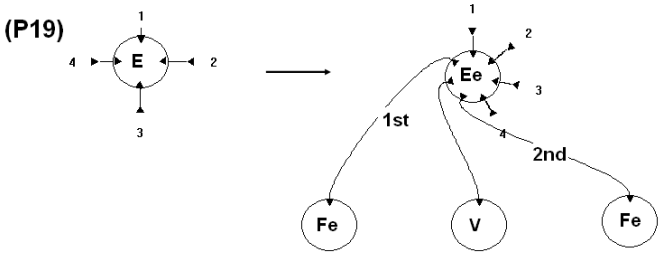


Fig. 12. Production for breaking an element edge

is adjacent to the boundary. This is expressed by **(P16)** productions in Fig. 11. Analogous productions are utilized for mixed rectangular / triangular neighbors. The productions allows for breaking an element edge, by checking adjacent element interior. If two adjacent interiors are broken, the label of the element edge is changed from **F** to **E**. Only element edges denoted by **E** label can be broken. The physical breaking of an element edge is executed by graph grammar productions **(P19)** presented in Figure 12. An element interior can be broken only if all adjacent edges are of the same size as the interior. This is expressed by **(P39)** production in Fig. 13.

In other words, the history of refinements for adjacent edges is coded within the label of graph vertex representing element interior. The interior can be broken only after breaking adjacent edges and propagating the adjacency information along the refinement trees. The goal of the mesh regularity rule is to avoid multiple constrained edges, which leads to problems with approximation over such edges. The mesh regularity rule enforces breaking of large adjacent unbroken elements before breaking small element for the second time, which is illustrated

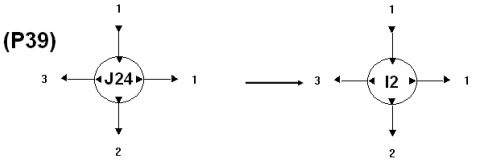


Fig. 13. Production allowing for breaking an element interior

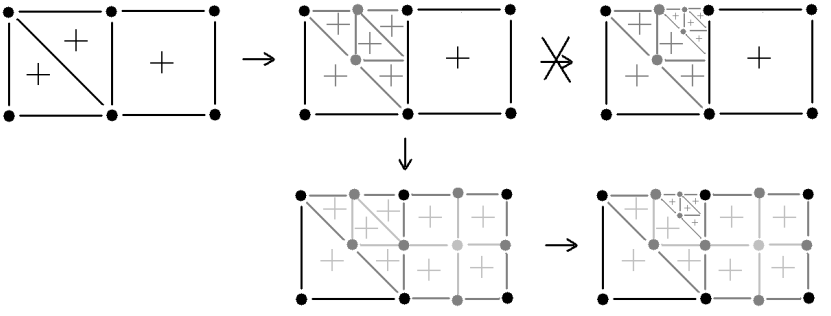


Fig. 14. Enforcement of the one order irregularity mesh rule

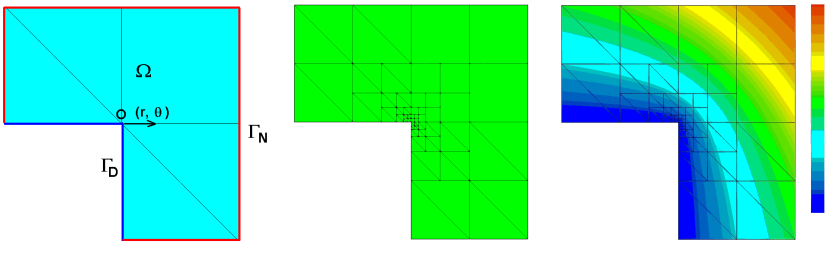


Fig. 15. The initial mesh, the last mesh with 1% relative error of the solution, and the solution to the Fichera model problem

in Fig. 14. The mesh regularity rules are enforced on the level of graph grammar syntax.

3 Numerical Example

We conclude the presentation with the sequence of triangular finite element meshes generated for the L-shape domain model problem [8]. The problem consists in solving the Laplace equation $\Delta u = 0$ in Ω over the L-shape domain Ω presented on left panel in Fig. 15. The zero Dirichlet boundary condition $u = 0$ is assumed on the internal part of the boundary Γ_D. The Neumann boundary

condition $\frac{\partial u}{\partial n} = g$ is assumed on the external part of the boundary Γ_N. The temperature gradient in the direction normal to the boundary is defined in the radial system of coordinates with the origin located in the central point of the L-shape domain $g(r,\theta) = r^{\frac{2}{3}} sin\frac{2}{3}\left(\theta + \frac{\pi}{2}\right)$. The solution $u : R^2 \supset \Omega \ni u \mapsto R$ is a temperature distribution inside the L-shape domain. The initial mesh consists in 1 rectangular element and 4 triangular elements, see Fig. 17. The self-adaptive *hp*-FEM generates a sequence of meshes delivering exponential convergence of the numerical error with respect to the mesh size. The initial mesh, the last mesh with 1% relative error, and the solution, are presented in Fig. 17.

4 Conclusions

The CP-graph grammar is the tool for a formal description of mixed triangular and rectangular mesh transformations utilized by adaptive FEM. It models all aspects of the adaptive computations, including mesh generation, h and p refinements, as well as mesh regularity rules, including elimination of multiple constrained nodes. The technical nightmare with implementing the mesh regularity rules has been overcome by including the mesh regularity rules within the graph grammar syntax. The graph grammar have been formally validated by utilizing graph grammar definition software [5]. Our future work will concern several extensions of the presented CP-graph grammar. First, the graph grammar will be extended to model three dimensional meshes with tetrahedral, hexahedral, prism and pyramid elements, with an isotropic mesh refinements. Second, both the two and three dimensional graph grammars will be extended to support anisotropic mesh refinements. Moreover, the graph grammar will be extended to model the execution of the solver over the computational meshes, in three dimensions, for different kinds of elements.

Acknowledgments. The work of the second author has been supported by the Polish Ministry of Scientific Research and Information Technology and by the Foundation for Polish Science under Homming Programme.

References

1. Beal, M.W., Shephard, M.S.: A General Topology-Based Mesh Data Structure. International Journal for Numerical Methods in Engineering 40, 1573–1596 (1997)
2. Flasiński, M., Schaefer, R.: Quasi Context Sensitive Graph Grammars as a Formal Model of Finite Element Mesh Generation. Computer Assisted Mechanics and Engineering Science 3, 191–203 (1996)
3. Grabska, E.: Theoretical Concepts of Graphical Modeling. Part One: Realization of CP-Graphs. Machine Graphics and Vision 2(1), 3–38 (1993)
4. Grabska, E.: Theoretical Concepts of Graphical Modeling. Part Two: CP-Graph Grammars and Languages. Machine Graphics and Vision 2(2), 149–178 (1993)
5. Grabska, E., Hliniak, G.: Structural Aspects of CP-Graph Languages. Schedae Informaticae 5, 81–100 (1993)

6. Paszyński, M., Paszyńska, A.: Graph transformations for modeling parallel *hp*-adaptive Finite Element Method. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Wasniewski, J. (eds.) PPAM 2007. LNCS, vol. 4967, pp. 1313–1322. Springer, Heidelberg (2008)
7. Paszyńska, A., Paszyński, M., Grabska, E.: Graph transformations for modeling *hp*-adaptive Finite Element Method with tringular elements. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 604–613. Springer, Heidelberg (2008)
8. Demkowicz, L.: Computing with hp-Adaptive Finite Elements. Chapman & Hall/Crc Applied Mathematics & Nonlinear Science, vol. I (2006)
9. Paszyński, M., Kurtz, J., Demkowicz, L.: Parallel Fully Automatic hp-Adaptive 2D Finite Element Package. Computer Methods in Applied Mechanics and Engineering 195(7-8)(25), 711–741 (2006)

Agent-Based Environment for Knowledge Integration

Anna Zygmunt, Jarosław Koźlak, and Leszek Siwik

Department of Computer Science
AGH University of Science and Technology, Kraków, Poland
{azygmunt,kozlak,siwik}@agh.edu.pl

Abstract. Representing knowledge with the use of ontology description languages offers several advantages arising from knowledge reusability, possibilities of carrying out reasoning processes and the use of existing concepts of knowledge integration. In this work we are going to present an environment for the integration of knowledge expressed in such a way. Guaranteeing knowledge integration is an important element during the development of the Semantic Web. Thanks to this, it is possible to obtain access to services which offer knowledge contained in various distributed databases associated with semantically described web portals. We will present the advantages of the multi-agent approach while solving this problem. Then, we will describe an example of its application in systems supporting company management knowledge in the process of constructing supply-chains.

1 Introduction

The accessibility of different knowledge bases and especially the Internet network in its current form at present provides a huge amount of different information resources and numerous application services. An important challenge here is to guarantee the most efficient and user-friendly solution of the possibilities provided by this kind of environment.

This may be offered by the development of a Semantic Web that embraces information existing in the WWW network and by integrating it with a software in order to realize various activities in response to the demands and requirements of the user, taking into consideration their preferences and cooperating with such different modules distributed in the Internet. An elaboration of this kind of infrastructure will offer a new quality of Web use and will provide users a greater amount of knowledge and services that are described in a correct way and useful to them. To achieve this goal, it is necessary to solve many different problems at the same time: a representation of a semantic of web resources, to guarantee reasoning mechanisms to supplement the lack of or only partial knowledge, the composing of web services, an analysis of the knowledge gathered and what is available in different thematic portals using techniques of artificial intelligence such as data mining or machine learning.

We will focus particularly on presenting an infrastructure needed for leading the process of integration of knowledge expressed with the use of knowledge description languages [8,9] like OWL [5] and RDF [1]. We will present different kinds of knowledge integration, described in the literature. Then we will present an environment for

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 885–894, 2009.

integrating ontologically expressed knowledge. Special attention will be given to the application of a multi-agent approach to solve this problem. We will also present our pilot environment based on the multi-agent platform JADE.

2 Study of Problems

In our work we are going to focus on three important problems regarding the knowledge expressed using ontologies: reasoning, integration and application of the multi-agent process to perform the integration of the knowledge stored in distributed knowledge bases. In the following sub-chapters we will outline in short the characteristics of research carried out in these domains.

2.1 Ontology Reasoning

The fundamental principle of the reasoning process in expressing the knowledge using ontologies is Description Logics (DL) [10,3] which define two knowledge components: TBox and ABox. The TBox (Terminology Box) component includes domain definitions (i.e. declarations of concept properties). TBox determines the subsumption relations and is implementation-independent. The knowledge stored in TBox can be characterised as persistent (it does not evolve over time). The ABox (Assertional Knowledge) component contains case-specific knowledge: it assigns specific meanings to concepts derived from TBox. The knowledge stored in ABox is inherently transient and dependent on circumstances. The basic reasoning mechanism which exists in ABox involves checking whether a given unit is included in a selected concept. More complex mechanisms exist as well although they all follow on from the basic one.

2.2 Ontology Integration

As a result of integration, there is the possibility to obtain access to higher amounts of information, it is also possible to obtain additional information which results from relations between concepts present in different knowledge sources. It seems that an integration of ontologies may provide additional qualities to the applications being developed. Unfortunately at present, there are no ready-to-use, simple and in-use solutions that could automatically control this process. It is possible to distinguish [12] several different schemes of ontology integration. In the single ontology approach, all the ontologies are integrated into one global one and a unified access to a knowledge model takes place. In the multiple ontology approach, each of the information sources possesses their own local ontology, which makes it possible to use a separate dictionary and the ontologies may be developed independently.

A combination of these two approaches is a hybrid approach where its information source has its own local ontology derived from a global ontology to ensure easier adjustment. Carrying out the ontology integration the following operations are usually used [7]: mapping, alignment merging and integrating. During the process of ontology integration the following elements of the problem solution exists: an analysis of the lexical and structural similarities, in the aim of finding concepts most appropriate to

one another, a use of existing tools and knowledge bases to find the relations between the concepts and domains to which they can be rated. It is worth mentioning here that semantic dictionaries like WordNet are specially designed high-level ontologies.

Upper level ontology is an attempt to create a general description of concepts and relations among them which may be the same and possible to use for different knowledge domains. The main goal of the creation of such a knowledge base is to make it possible to access different ontologies through the upper ontology. One of the most popular definitions of the approach based on upper ontologies is SUMO (Suggested Upper Merged Ontology) [11].

2.3 Overview of Environments for Ontology Integration

To make possible the efficient use of the knowledge included in distributed knowledge sources, it is necessary to possess an appropriate infrastructure. In our opinion, the multi-agent [6] approach offers important advantages which support a knowledge integration process. The multi-agent approach is based on the assumption that systems have a distributed and decentralised structure consisting of autonomous rational entities called agents that cooperate with one another to realise their own tasks. This subsequently results in the realisation of the global goal of the system. Additionally, agents are equipped with features such as: perception of the environment, capacity to perform actions which modify the environment, use of interaction protocols which describe the possible conversation flow between them and using agent communication languages which describe the structure of exchanged messages, the construction of plans that have as their goal the realisation of assumed goals and governing the process of the machine learning so that it can realise its task in the dynamically changing environment as best as possible. These features may be very useful when we create an infrastructure for knowledge integration.

Several multi-agent environments were performed that have the goal of support in the process of agent development, deployment and multi-agent interactions. The set of specifications FIPA has the goal of elaborating requirements which agents have to fulfil to make the cooperation among them possible (http://www.fipa.org). Particularly, these specifications contain multi-agent communication languages and protocols, agent-transport protocols, agent management rules, agent architectures and their application. The most popular agent platform is JADE (Java Agent Development framework) [4]. During the realisation of the system other tools and software for use of ontologies were applied: ontology editor Protege, a library for accessing ontology models JENA and to create mappings between the ontology model and information stored in the database (D2RQ).

3 Environment for Ontology Integration

Our work concerns an overview of problems which we encountered during the development of an infrastructure that supports reasoning using a semantic knowledge representation distributed in the Internet while taking advantage of the possibilities of different actions on the basis of knowledge described in such a way. This knowledge was described with the use of ontological description languages (like OWL).

In the presented solutions, we will focus our attention on the application of different integration techniques: working on the local level (concerning particular concepts and their attributes) algorithms of lexical and structural comparison or checking of similarity between larger parts of a graph with the use of Similarity Flooding algorithm. We also applied additional approaches based on a thesaurus for looking for synonyms or on the use of high level ontology - Upper Ontology (such as SUMO) to adjust concepts from the ontology to a given set of concepts which identify important notions.

The infrastructure takes advantage of the agent platform JADE. The agent infrastructure for ontology integration was based on the assumption that a knowledge expressed using ontology languages is accessible in the form of decentralized knowledge sources 1. The system has a distributed architecture, each node possess an program instance and own ontology with a database. We can distinguish three main kinds of agents: Container Agent (represents an node), DistributedQuery Agent (represents the queries sends to the system by users) and Distributed QueryAgent (supervises a ontology integration process on its node).

Such architecture has many advantages: the possibility of information exchange between different centres, lack of necessity of possessing a knowledge about the instances of application currently accessible and the free flow of knowledge.

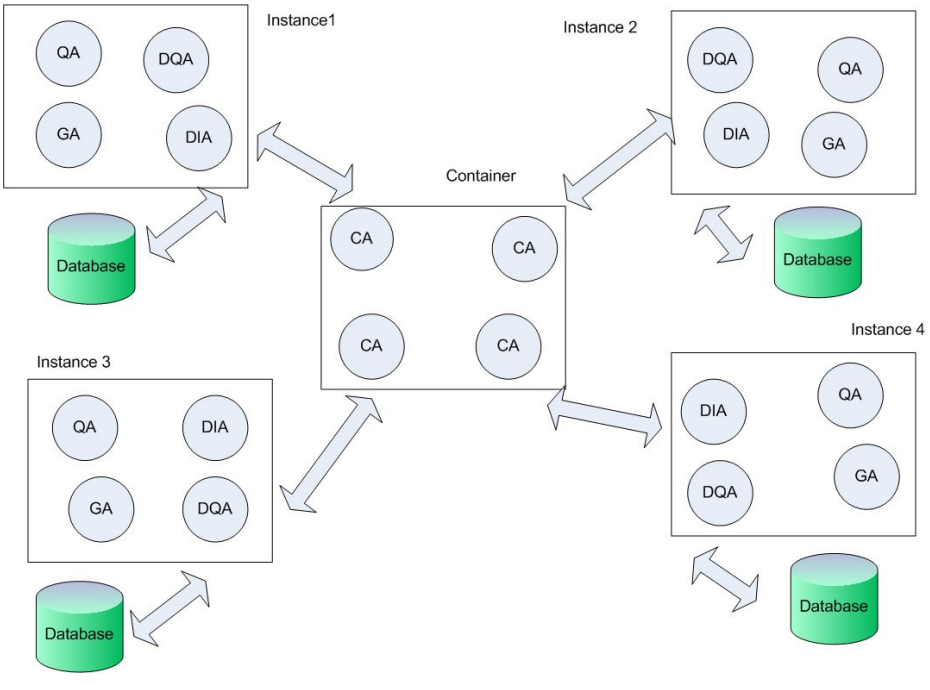


Fig. 1. General system architecture (CA - Container Agent, DIA - Distributed Integration Agents, QA - Query Agent, GA - Grade Agent)

There are also disadvantages of this pilot realization. The costs of communication between the agents are high, during the integration phase the number of messages increases linearly with the increase of agent number.

3.1 Agents in the System

The agents are inheriting the functionality provided by AbstractAg class defined in JADE environment: There are following Agents in the System:

- MasterAgent—performs the necessary operations during program start, like the creation of other agents,
- GradeAgent—is an agent which performs integration using different methods, it also makes a choice of the most appropriate integration methods for a given problem and then sends a request to a special agent,
- QueueAgent—an agent which puts the integration requests into the queue, " ContainerAgent - an agent which creates a distributed infrastructure, it makes the communications between the instances of the program possible,
- DistributedIntegratingAgent—after receipt of the query it sends its ontology to agent container,
- DistributedQueryAgent—this agent is created after the arrival of integration request, queries to ontology with the use of a functionality of JENA language, sends a requests to deliver ontologies for integration to all the needed system instances and then after receiving the ontologies, it sends them to QueueAgent,
- IntegratingAgent—integrates ontological model being sent to it, provides the main functionality for each integration agent, the special functionalities are provided by agents using a particular integration scheme,
- MetricSimilarityIntegratingAgent—compare instances using similarity metric,
- a set of different integrating agents, each of them is functioning according to its own algoriths, which use methods presented in section about integration methods: PromptIntegrating Agent (uses functionality offered by a Prompt tool), SimilarityIntegratingAgen , JenaIntegratingAgent (uses basic functions offered by Jena library), DictionaryIntegratingAgent (uses a dictionary of synonyms and looks for the suitable synonyms in the ontologies being integrated), and the last three using extention integrating methods - InstanceIntegratingAgent, InstanceSymmetricIntegratingAgent, InstanceJaccardIntegratingAgent.

The integration of ontologies is coordinated by an agent called GradeAgent. This agent supervises a process of estimation of similarity of given classes/concepts with the use of different methods. It creates also another integrating agents, sends to them messages with models for evaluation, receives matrixes with results and constructs integration commands as well as initialises a final step of integration.

3.2 Interactions between Agents

The realised environment is based on the cooperation of many agents that find things out from each other thanks to a repository. Container Agents are run on the server, their

role is to represent original instances of the program. During the start of each instance two agents: Container Agent and Queue Agent (responsible for queuing requests for ontology integration coming from other instances) are created. At the moment of a request arriving, the following operations are performed:

1. The agent DistributedQueryAgent is created.
2. The DistributedQueryAgent verifies in the repository, how many ContainerAgents are present and gets their list. A request is sent to all considered instances to make their ontologies accessible for the integration process.
3. ContainerAgent in the moment of receiving a request from DistributedQueryAgent creates in its original instance the DistributedIntegratingAgent (E) and transfers a request to it.
4. DistributedIntegratingAgent picks the ontology stored in its own origin instance Then it sends it back to the AgentContainer.
5. ContainerAgent prepares a new message which contains the obtained ontology and sends it to a proper DistributedQueryAgent.
6. DistributedQueryAgent sends subsequent incoming ontologies to QueueAgent which places them into the queue.
7. The integration is executed. Only one integration at the same time may be performed, to guarantee the coherence of the knowledge. Ontologies present in the queue are subsequently sent to the created GradeAgent which, on the basis of the ontology stored in its own instance and obtained in the message, performs an estimation and performs the integration algorithms.
8. The last queue sends a confirmation by GradeAgent which signifies the end the integration and allows then to start the next integration process by taking the ontology from the queue managed by QueueAgent.

3.3 Integration Process

The process of integration (Fig. 2) consists of the following steps:

- Receiving of information containing the ontology models. After its initialisation an agent is in the state of waiting for information which orders it to start the process of estimation and integration. After receiving this information, an agent unpacks it and obtains the ontology models and a queue which should be performed by it.
- Initialisation of agents needed to perform estimations and integration. During the initialisation of GradeAgent it obtains a list of agents, which should be used. On the basis of this list, they are initialised and waiting for the orders of GradeAgent.
- Estimation procedure - a next step is to order the agents to estimate the similarities between concepts/classes. All possible combination of classes from both considered ontologies are checked with the use of the selected methods.
- Each integrating agent sends a matrix with the evaluations of similarities. The Grade Agent is looking for the best adjusted classes in the matrixes. As a results it obtains a list of integration commands describing the mode of integration of given concepts (copying or merging).
- On the basis of the list of commands a final integration process is performed. As a result, an integrated model is obtained.

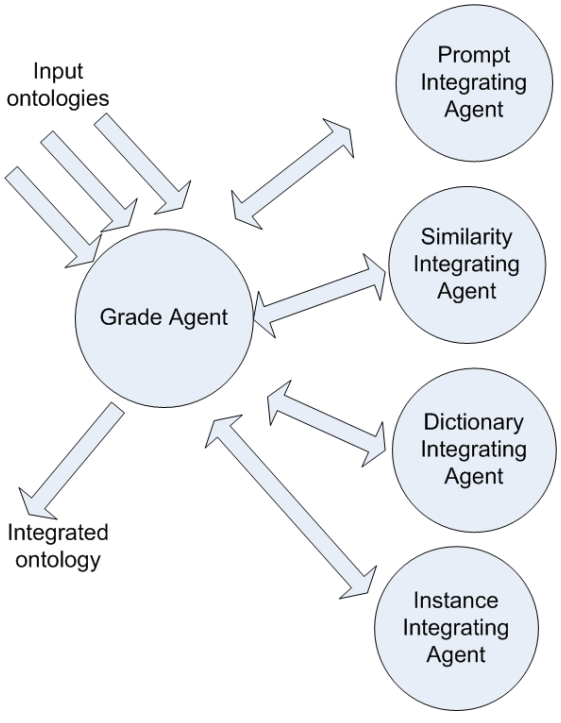


Fig. 2. Integration process

4 Example of Application Domain: Supply Chains Management

Ontology-based approaches also seem to be a perfect solution for modelling and optimizing production processes. The amount of participants of such processes (customers, suppliers, producers etc) as well as a complexity and depth of relations between all participants of such processes, the number of parameters, coefficients and factors that have to be taken into consideration and finally all the above, causes supply chain management and optimization, production process management and optimization to be still a challenging task for contemporary algorithms, tools, representations and methods. The authors attempted to apply such ontology-based processes for modelling and optimization and below, one of the proposed and preliminary assessed approaches presented and discussed in short.

4.1 Ontologies

The developed system operates on the knowledge describing products, producers and orders. Logically, it can be separated into the part describing factories and products and into the part describing orders. With both of these parts there is a separated ontology associated. In order to realise the goal included in our research, the system is responsible for searching factories being able to produce commodities in a expected time, price,

quality etc or for searching for substitutes of given product(s) if necessary and a factory being able to produce it.

As one of our goals was also to present the ability of integrating the domain-consistent data - the idea of dividing ontology into two separated ontologies is even more so justified.

The ontology describing factories and products includes the following information: the hierarchy of factories, categorisation of products; detailed hierarchy of products; localization of factories, information about products stored in database, qualitative and quantitative specifications of semi-products of a given product.

Next, the ontology describing orders includes the following information: the hierarchy of products and information about orders. In the proposed system, orders are visualized as an individual class of order. They are described by the orders group of parameters. With each order there is the number of ordered units associated. Next, ordered units are presented in the system as instances of appropriate class.

4.2 Architecture

From the architectural point of view there can be distinguished three main modules in the system (Fig. 3): model description module; data module; operation performing module. Model description module enables and is responsible for adding information about individuals coming from data module (with the use of JENA library). It is able for operating on any ontology but in presented system and application it includes obviously mentioned ontologies describing factories, commodities and orders.

Data module includes information about individuals representing concepts from model description module. It is realised as a database implementation (with the use of MySQL server). The operation of mapping between ontology and database notions can be realised with the use of many different databases and it is realised thanks to D2RQ-based approach.

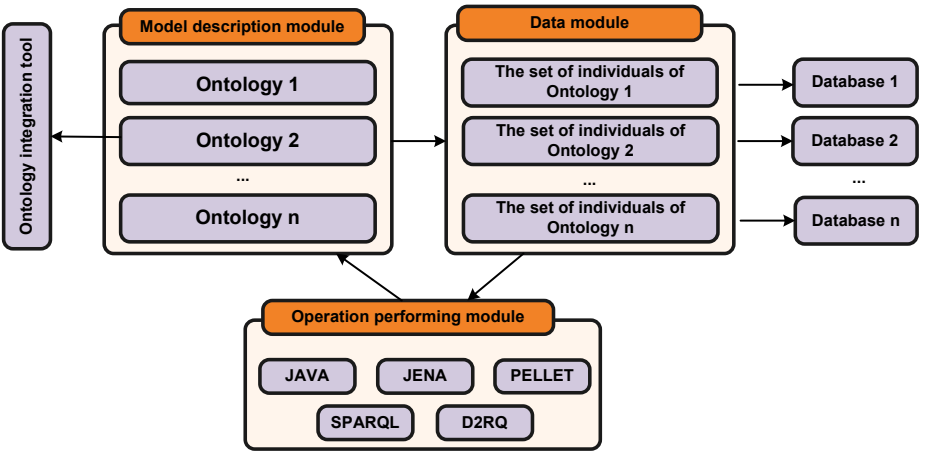


Fig. 3. High-level system architecture

Operation performing module supplies input and output for any operation performed by the system. It is implemented in JAVA and it realizes reasoning on the data set coming from model description module with the use of mechanisms supplied by PELLET [2] library. Operation performing module uses the interface of performing queries of SPARQL supplied by JENA library. It also supplies the mechanism of D2RQ by JENA API. Data coming from model description module are transformed to the JENA model and next with the use of PELLET library it is possible to have access to the processed and reasoned data.

4.3 Discussion of Results

One of the aspects of our research was to assess the ability, flexibility, simplicity and efficiency of knowledge integration. As mentioned before, ontologies used in our model were divided into two separate ontologies describing factories/products and orders respectively. What we wanted to do was to assess if their integration is possible and if so - how easy and how efficient or how difficult it is. During performed experiments dedicated tool developed in our group was used. The tool can operate in two modes: simple and full mode. In simple mode the user is able to define ontologies to integrate as well as some basic options - i.e. the user has to define if ontologies that should be integrated are dissimilar both lexically and structurally, if they are similar lexically but dissimilar structurally, if they are similar structurally and dissimilar lexically and finally if they are similar both: lexically and structurally. In full mode the user has to define among the others: the measure of the similarity (lexical, structural, global etc.), the algorithm and the filter of the similarity that should be used.

With the use of the above-mentioned tool we tried to integrate ontologies describing factories and products and order.

To estimate the quality of the performed integration measures of unconditional (defined as ratio of correct adjustments performed by the application to expected correct adjustments) and conditional quality success (defined as a ratio of correct adjustments performed by the application to all obtained adjustments). As a result of the considerable similarity of the subclasses of the Factories and Products Classes which have convergent names and similar internal structure, excessive adjustments are present.

5 Conclusions

In this work, an overview of methods of ontology integration and an agent infrastructure were presented, which uses these methods to integrate distributed knowledge sources. As an integration algorithm a similarity flooding was chosen since it allows for distinguishing points where classes that are being integrated seem to be similar. Almost all expected matchings were realized. It turned out also that using a similarity flooding algorithm was an appropriate choice since products are aggregated in both ontologiess inside one super-class and as a result, more appropriate matchings were realized Then we presented a domain of the application of the ontological approach, emphasizing its advantages (on the level of reasoning and integration) which it offers during analysis and which is necessary in specific problems and for specific domain knowledge.

Acknowledgements. We would like to thank the former student of Computer Science from AGH-UST, especially Karol Hadała, Krzyszof Kosiński, Krzysztof Łosiński, Adam Łuszpaj and Kamil Szymański, for their participation in the realisation of presented systems.

References

1. RDF Vocabulary Description Language 1.0: RDF Schema (2003), `http://www.w3.org/TR/PR-rdf-schema`
2. Pellet: The Open Source OWL DL Reasoner (2008), `http://clarkparsia.com/pellet`
3. Baader, F., Nutt, W.: The Description Logic Handbook. In: Basic Description Logics. Cambridge University Press, Cambridge (2002)
4. Bellifemine, F., Poggi, A., Rimassa, G.: Developing Multi-agent systems with a FIPA-Compliant Agent Framework. Software Practice and Experience 31(2), 102–128 (2001)
5. Dean, M., Schreiber, G.: OWL Web Ontology Language Reference. W3C Working Draft (2003), `http://www.w3.org/TR/owl-ref`
6. Ferber, J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison Wesley Longman, Amsterdam (1999)
7. Fridman, N., Musen, M.: SMART: Automated Support for Ontology Merging and Alignment. In: Twelth Workshop on Knowledge Acquisition, Modeling, and Management, Banff, Canada (1999)
8. Kozlak, J., Zygmunt, A., Luszpaj, A., Szymanski, K.: The Process of Integrating Ontologies for Knowledge Base Systems. In: Software engineering: evolution and emerging technologies, Frontiers in Artificial Intelligence and Applications, pp. 259–270 (2005)
9. Luszpaj, A., Zygmunt, A., Kozlak, J., Nawarecki, E.: A concept, implementation and testing of an algorithm for knowledge integration using upper ontology. In: Proc. of the 16th International Conference on Systems Science. Decision support and expert systems, Wroclaw, Poland, pp. 397–404 (2007)
10. Nardi, D., Brachman, R.J.: An Introduction to Description Logics. In: The Description Logic Handbook. Cambridge University Press, Cambridge (2002)
11. Niles, I., Pease, A.: Towards a standard upper ontology. In: Proceedings of the International Conference on Formal Ontology in Information Systems. ACM, New York (2001)
12. Wache, H., Vgele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., Hubner, S.: Ontology-based integration of information-a survey of existing approaches. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI 2001), Workshop: Ontologies and Information Sharing, Seattle, USA (2001)

Agent Strategy Generation by Rule Induction in Predator-Prey Problem

Bartłomiej Śnieżyński

AGH University of Science and Technology,
Department of Computer Science Kraków, Poland
Bartlomiej.Sniezynski@agh.edu.pl

Abstract. This paper contains a proposal of application of rule induction for generating agent strategy. This method of learning is tested on a predator-prey domain, in which predator agents learn how to capture preys. We assume that proposed learning mechanism will be beneficial in all domains, in which agents can determine direct results of their actions. Experimental results show that the learning process is fast. Multi-agent communication aspect is also taken into account. We can show that in specific conditions transferring learned rules gives profits to the learning agents.

Keywords: multi-agent systems, rule induction, machine learning.

1 Introduction

Decentralized problem solving becomes more and more popular. Architecture, which can be used for this purpose is multi-agent system. The problem is that in complex environments it is very difficult (or sometimes impossible) to specify and implement all system details a priori. A solution is a learning method application, which adopts the system to the environment. Learning can be also useful if the environment is not stationary.

To apply learning in a multi-agent system, one should chose a method of learning. There are many algorithms developed so far. However; in multi-agent systems most applications use reinforcement learning.

In this paper we show that rule induction can be successfully applied to generate agent strategy in the predator-prey domain, in which reinforcement learning and evolutionary computation methods were applied.

Rule induction is a supervised learning method, therefore, in the contrast to the methods mentioned above, it needs labeled examples (e.g. percepts-action pairs) to generate knowledge. We show, that in this domain an agent is able to generate such examples. The reason is that results of actions performed by learning agents are visible immediately. Any supervised learning method can be applied in a similar way in all environments with such a property.

In the following sections, related research considering learning in multi-agent systems is briefly discussed, the developed system is described, and experimental results are presented and analyzed.

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 895–903, 2009.

2 Learning in Multi-agent Systems

Good survey of learning in multi-agent systems can be found in [1] and [2]. As it was mentioned above, the most popular learning technique used is reinforcement learning that allows to learn agent strategy: what action should be executed in a given situation. Other techniques can be also applied: neural networks, models coming from game theory as well as optimization techniques (like the evolutionary approach, tabu search etc.). However, optimization techniques improve performance of the system using many populations of agents instead of a single agent.

Reinforcement learning allows to generate a strategy for an agent in a domain, in which the environment provides some feedback after the agent has acted. Feedback takes the form of a real number representing reward, which depends on the quality of the action executed by the agent in a given situation. The goal of the learning is to maximize estimated reward. This method was successfully applied in the predator-prey domain [3]. In this work predator agents use reinforcement learning to learn a strategy minimizing time to catch a prey. Additionally, agents can cooperate by exchanging sensor data, strategies, or episodes. Experimental results show that cooperation is beneficial. Other researchers working on this domain successfully apply genetic programming [4] and evolutionary computation [5].

There is only a few works known to the author on supervised learning in multi-agent systems. Rule induction is used in a multi-agent solution for vehicle routing problem [6]. However; in this work learning is done off-line. First, rules are generated by AQ algorithm (the same as used in this work) from traffic data. Next, agents use these rules to predict traffic. In [7], agents learn coordination rules, which are used in coordination planning. If there is not enough information during learning, agents can communicate additional data during learning. Airiau [8] adds learning capabilities into BDI model. Decision tree learning is used to support plan applicability testing.

In [9] there is a comparison of supervised learning and reinforcement learning methods in a Fish-Banks game. Agents run fishing companies and learn how to allocate ships. In this case, agents using rule induction perform slightly better than ones using reinforcement learning.

Universal architecture for learning agent can be found in [10]. It fits mainly reinforcement learning. Sardinha et. al, propose a learning agent design pattern, which can be used during system implementation [11]. More abstract architecture, which is used in this work, is presented in [12].

3 System Architecture

3.1 Predator-Prey Domain

Predator-prey domain is a simple simulation with two types of agents: predators, and preys. The aim of a predator is to hunt for a prey. Environment is a grid world with size $n \times n$ with "glued" opposite edges (it is a torus). Time is discrete,

its flow is represented by turns. In every turn all agents receive percept data from the environment and chose their actions, which are next executed.

Agents can move in the four directions (up, down, left, and right) or not move at all. If predator occupies a field next to the prey, the prey is captured.

Predators have limited range of sight. If prey is closer than a given threshold, predator gets information about type of the prey, and its relative position. If more then one prey is in range, only the closer one is visible.

Two types of the preys are defined: bird and mouse. First one moves up, down or does not move at all with equal probability. Similarly, mouse move left, or right, or does not move.

Two types of predators are defined: random and learning. The former moves in the four directions or do not move with equal probability. The latter uses rule induction to improve performance.

3.2 Learning Predator

Learning predator architecture is presented in Fig. 1. In this application *percepts* received by the agent is nul if no prey is in the observation range, or is a triple (t, dx, dy), where t is a type of a prey, and dx, dy are relative coordinates of the prey along X and Y axis, respectively.

Initially, *processing module* selects actions from the set {none, up, down, left, right} randomly. If, after actions execution, distance to the prey decreases, new example is stored in the *training data* memory. Percept triple from the previous round form attribute values of the example and action executed is its class. Here important property of the environment is used. It is possible to generate training data because action results are immediately visible.

Rules generated during learning have a form $p_1, p_2, \ldots, p_n \rightarrow a$, where p_i are tests on the attributes representing current percepts, and a is an action, which should be executed.

During a learning phase training data is sent to the *learning module*. In this research AQ21 rule induction program is used [13]. It is the latest implementation

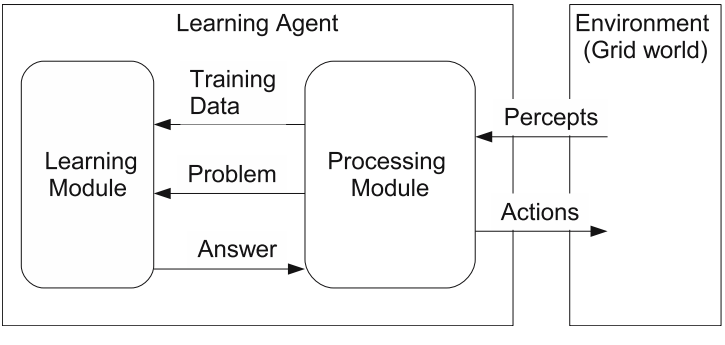


Fig. 1. Learning agent architecture

of AQ algorithm [14]. Rules are generated using sequential covering: the best rule (e.g. giving correct answer for the most examples) is constructed by a beam search, examples covered by this rule are eliminated from the training set, and the procedure repeats.

AQ21 program is executed in a theory formation mode. It means that rules generated are sound and complete. Ambiguity option is set to `IncludeInMajority`. If there are more events in the training data with the same values of all attributes and different classes (actions), learning algorithm assigns to the event the majority class. All other options are set to their default values.

In the future various option settings can be tested. However; it is one of advantages of rule induction application for agent strategy generation: tuning of parameters is not necessary to get good results.

If knowledge is generated, processing module can use learning module to chose an action in the current situation. Input to the learning module is a *problem*, which consists of a percept triple, and output – *answer* is a conclusion of the rule matching the percepts or nul if the rule is not found (in such a case agent executes random action).

In this research two types of communication were considered for the predator agent: exchange of generated rules and exchange of the training data. If no matching rule is found, predator agent can ask another agent for a matching rule and store it in its knowledge base. Exchange of the training data means that during learning agent asks for examples stored in other agent's memory to generate better knowledge. During experiments only the former case was tested. Results are initial, but we can show that in specific conditions transferring learned rules gives profits to the learning agents.

3.3 Implementation

The software used in experiments is written in Prolog, using Prologix compiler [15]. Prologix is an extension of BinProlog that has many powerful knowledge-based extensions (e.g. agent language LOT, Conceptual Graphs and KIF support). AQ21 program [13] is executed from Prolog and rules generated are added to the code as additional horn clauses.

4 Experimental Results

4.1 Setup

Let us introduce several definitions. Game is defined as a sequence of turns beginning with the initial positions of agents and ending in a turn when all preys are captured. Twelve consecutive games is called a sequence. It is assumed that memory and knowledge base of learning predators is kept unchanged between games in a sequence. However, it is cleared between sequences. Performance of predators is measured by a number of turns in a game; the less, the better.

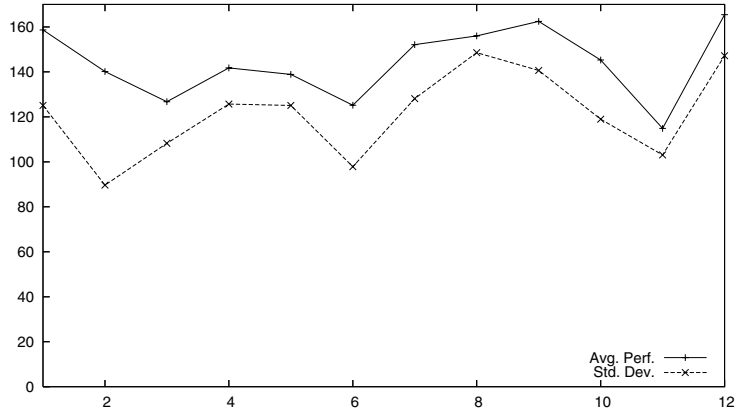


Fig. 2. Average performance and its standard deviation for random agents in consecutive games

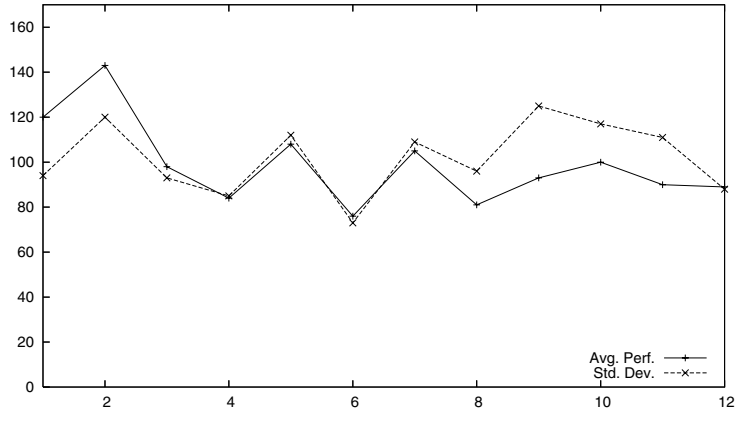


Fig. 3. Average performance and its standard deviation for learning agents (without communication) in consecutive games

Grid world in experiments has dimensions 16×16. Initial positions of predators are (0, 0) and (8, 8) to maximize distance between them. There are two prey agents: bird and mouse. In all experiments 60 sequences of games are executed and performance measures in every game are stored. Learning predators execute learning algorithm at the end of every even game.

4.2 Random Predators

The aim of the first experiment is to test performance of random predators. Two of them take part in the games. Initial positions of preys are random. The average performance of the predators is 140.01. Its standard deviation is

very high: 120.50. Hence, as we can see in Fig. 2, average performance varies from game to game in the sequence.

4.3 Learning Predators

Two learning predators take part in this experiment. They do not communicate. Initial positions of preys are random. Average performance measures in consecutive games are presented in Fig. 3.

The learning process is fast. In this experiment after executing about 300 random actions in two games, and collecting examples, performance changes from about 130 to 90.

As we can see, performance of learning agents increases rapidly at the beginning of the learning process, when generated rules are used instead of a

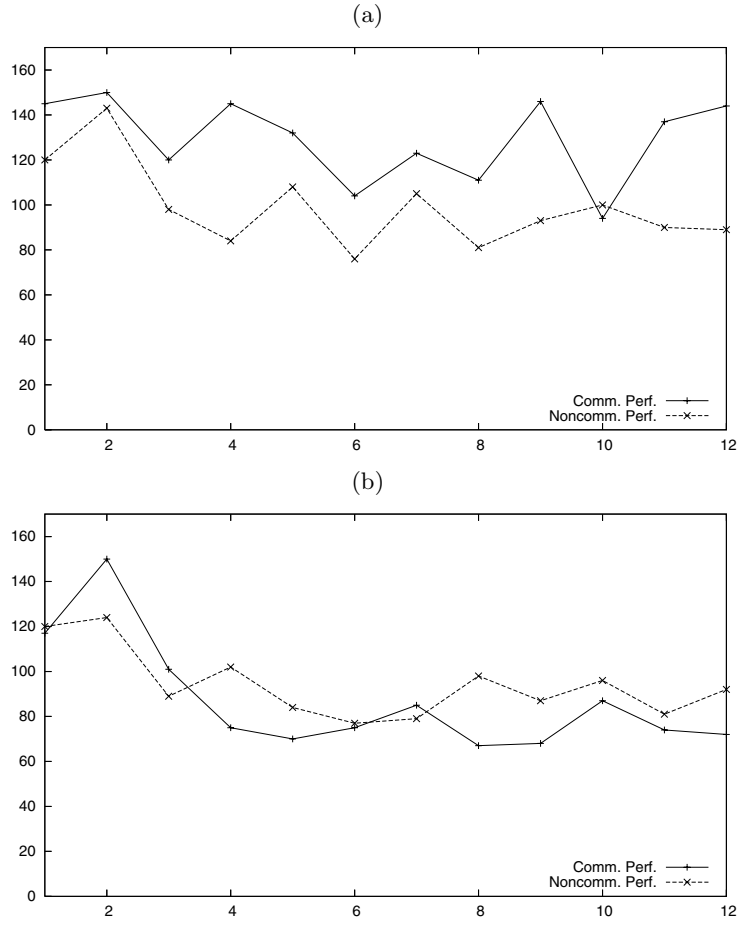


Fig. 4. Performance comparison or learning agents with and without communication in experiment with random initial prey positions (a) and fixed initial prey positions (b)

random choice. Next it increases slowly, because new examples do not contain any significant knowledge. The performance stabilizes at the end of the process.

4.4 Learning Predators with Communication

This experiment tests influence of communication. In Fig. 4-(a) we can see comparison of performances of communicating and non-communicating predators with random initial positions of preys. As we can see, communication not only does not help learning agents, but even makes the performance worse. Unfortunately, the reason of this behavior is not clear yet.

We tested another initial setting. In these experiments bird-prey starts in position (3, 0) with probability 0.8 and in (11, 8) with probability 0.2, and mouse-prey vice versa. In Fig. 4-(b) we can see performance comparison for this settings. Performance of communicating agents is better than ones without communication.

This result suggests that communication is profitable if a learning agent is rarely in the situation, which is common for the other agent. In such conditions the other agent can provide appropriate, useful knowledge to transmit and use.

4.5 Rule examples

Examples of rules learned by the predator-agent in a form of Prolog clauses are presented in Fig. 5. They can be interpreted in the following way. Both rules have action `left` in the conclusion. `S` is an identifier of a state, for which decision should be made. Rule (a) is applicable if the predator sees a bird which has the same or smaller by one x coordinate and relative vertical position is between -3 and 0. Premise of rule (b) checks if the predator sees a mouse, relative x coordinate is -2, and y is between -3 and -2.

(a)

```
dir(S,left) :-
    type(S,bird),
    dx(S,X),
    X >= -1,
    X =< 0,
    dy(S,Y),
    Y >= -3,
    Y =< 0.
```

(b)

```
dir(S,left) :-
 type(S,mouse),
 dx(S,-2),
 dy(S,Y),
 Y >= -3,
 Y =< -2.
```

Fig. 5. Examples of rules (in the form of Prolog clauses) learned by the agent

5 Conclusion and Further Research

In this paper idea of the agents using rule induction for learning strategy is presented. Solution proposed is tested on a specific example – predator-prey domain.

However, similar strategy generation works also for a Fish-Banks game [16]. The only condition for the environment to use this method is that agent should be able to observe direct results of its actions. In other words, results are not delayed and there is no reward assignment problem. We assume that rule induction, and more general, supervised learning can be applied in such circumstances in other domains.

Results for the Fish-Banks game [9] and initial results for the predator-prey problem [17] show that supervised learning give improvements faster than reinforcement learning.

Advantage of rule induction is clarity of rule-based knowledge representation. It is possible to interpret the knowledge base or check its correctness manually. It can be very important for some domains.

Additionally, because rule-based knowledge representation is modular, exchange of the knowledge is easy and has low cost. Only necessary rules can be transmitted.

A problem discovered in this research is a high value of standard deviation of performance measure in predator-prey domain. It makes this domain not very attractive for comparison of various methods. In the future research this domain should be modified and/or other domains will be explored.

Interesting issue for further research is developing agents using several learning methods for different aspects of their activity. Also exchange of training data as an another form of communication during learning should be tested. And last but not least, performance of rule learning agent should be compared to the performance of reinforcement learning agent in the same conditions.

Acknowledgments. The author is grateful to Arun Majumdar, Vivomind Intelligence Inc. for providing Prologix system (used for implementation), and for help with using it, Janusz Wojtusiak, MLI Laboratory, George Mason University for AQ21 software and assistance, and last but not least Jens Pfau, Technische Universität Darmstadt for suggesting predator-prey domain, and implementation of the first, Java version of the system using decision-tree induction and reinforcement learning.

References

1. Stone, P., Veloso, M.: Multiagent systems: A survey from a machine learning perspective. Autonomous Robots 8, 345–383 (2000)
2. Panait, L., Luke, S.: Cooperative multi-agent learning: The state of the art. Autonomous Agents and Multi-Agent Systems 11 (2005)
3. Tan, M.: Multi-agent reinforcement learning: Independent vs. cooperative agents. In: Proceedings of the Tenth International Conference on Machine Learning, pp. 330–337. Morgan Kaufmann, San Francisco (1993)
4. Haynes, T., Sen, I.: Evolving behavioral strategies in predators and prey. In: Adaptation and Learning in Multiagent Systems, pp. 113–126. Springer, Heidelberg (1996)
5. Giles, C.L., Jim, K.-C.: Learning communication for multi-agent systems. In: WRAC, pp. 377–392 (2002)

6. Gehrke, J.D., Wojtusiak, J.: Traffic prediction for agent route planning. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 692–701. Springer, Heidelberg (2008)
7. Sugawara, T., Lesser, V.: On-line learning of coordination plans. In: Proceedings of the 12th International Workshop on Distributed Artificial Intelligence, pp. 335–345, 371–377 (1993)
8. Airiau, S., Padham, L., Sardina, S., Sen, S.: Incorporating learning in bdi agents. In: Proceedings of the ALAMAS+ALAg Workshop (May 2008)
9. Śnieżyński, B.: Resource management in a multi-agent system by means of reinforcement learning and supervised rule learning. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4488, pp. 864–871. Springer, Heidelberg (2007)
10. Russell, S., Norvig, P.: Artificial Intelligence – A Modern Approach. Prentice-Hall, Englewood Cliffs (1995)
11. Sardinha, J., Garcia, A., Milidi, R., Lucena, C.: The agent learning pattern. In: Fourth Latin American Conference on Pattern Languages of Programming, SugarLoafPLoP 2004, Brazil (2004)
12. Śnieżyński, B.: An architecture for learning agents. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 722–730. Springer, Heidelberg (2008)
13. Wojtusiak, J.: AQ21 User's Guide. Reports of the Machine Learning and Inference Laboratory, MLI 04-3. George Mason University, Fairfax, VA (2004)
14. Michalski, R.S., Larson, J.: Aqval/1 (aq7) user's guide and program description. Technical Report 731, Department of Computer Science, University of Illinois, Urbana (June 1975)
15. Majumdar, A., Tarau, P., Sowa, J.: Prologix: Users guide. Technical report, VivoMind LLC (2004)
16. Śnieżyński, B., Koźlak, J.: Learning in a multi-agent approach to a fish bank game. In: Pěchouček, M., Petta, P., Varga, L.Z. (eds.) CEEMAS 2005. LNCS, vol. 3690, pp. 568–571. Springer, Heidelberg (2005)
17. Pfau, J., Śnieżyński, B.: Comparison of reinforcement and supervised learning in the predator prey game (2008) (unpublished)

Handling Ambiguous Inverse Problems by the Adaptive Genetic Strategy *hp*–HGS

Barbara Barabasz[2], Robert Schaefer[1], and Maciej Paszyński[1]

[1] Department of Computer Science
[2] Department of Modeling and Information Technology,
AGH University of Science and Technology,
Al. Mickiewicza 30, 30-059 Cracow, Poland
barabasz@metal.agh.edu.pl,
{schaefer,paszynsk}@agh.edu.pl

Abstract. We propose the new multi-deme, adaptive genetic strategy *hp*–HGS that allow solving ill posed parametric inverse problems that may posses multiple solutions. The strategy was obtained by combining *hp*-adaptive Finite Element Method with the Hierarchic Genetic Strategy. Its efficiency results from the coupled adaptation of the accuracy of solving optimization problem and the accuracy of *hp*–FEM direct problem solver. We present the simple L-shape domain benchmark that posses exactly two solutions. Test results show how the tunning of *hp*–HGS may affect on the number of solutions to be found. Moreover we discuss the artifacts that may appear by the particular setting of genetic operations.

1 Ambiguity of the Parametric Inverse Problems Solution

The inverse problems under consideration base on the particular class of direct problems:

$$\operatorname{ess\,min}_{u \in u_V} \{E(d;u)\}, \tag{1}$$

where $E(d;u) = \frac{1}{2} b(d;u,u) - l(u)$ stands for the total energy of the modeled physical system, V is the proper Sobolev space shifted by the Dirichlet boundary condition. The form of functionals b, l depend of the physical phenomena (e.g. linear elasticity, heat conduction) and of their parameters $d \in \mathcal{D}$, where $\mathcal{D}$ is the regular compact in $\mathbb{R}^N$, $N < +\infty$. We assume, that the above problem is well posed in the sense of Hadamard and its solution $u \in V$ can be obtained as the limit of the sequence $\{u_{h,p}\} \subset V$ of solutions of finite dimensional problems obtained by the *hp*–adaptive Finite Element Method (*hp*–FEM) assuming $h \to 0$ and $p \to +\infty$ (see Demkowicz [5]).

The inverse problem under consideration leads to encountering the unknown parameter $\hat{d} \in \mathcal{D}$ while the energy $J(\hat{d}) = E(\hat{d};u)$ of the exact solution $u \in V$ to (1) is known (e.g. it is measured during the laboratory test). Because the exact energy $J(d)$ is impossible to compute effectively for each $d \in \mathcal{D}$, we can only find the approximation $\hat{g} \in \mathcal{D}$ of the exact parameter $\hat{d} \in \mathcal{D}$ such that:

$$\lim_{h \to 0,\, p \to +\infty} \left| J_{h,p}(\hat{g}) - J(\hat{d}) \right| \leq \lim_{h \to 0,\, p \to +\infty} \left| J_{h,p}(g) - J(\hat{d}) \right| \quad \forall g \in \mathcal{D} \tag{2}$$

G. Allen et al. (Eds.): ICCS 2009, Part II, LNCS 5545, pp. 904–913, 2009.

where $J_{h,p}(d) = E(d; u_{h,p})$ stands for the energy of the approximate solution $u_{h,p}$ obtained for the parameter d.

The inverse problem presented above is a kind of the global optimization one with the very costly objective $\left|J_{h,p}(g) - J(\hat{d})\right|$. The most frequent difficulty is cosed by the ambiguity of its solution (multiple solutions) manifested as the objective multimodality. Moreover, the uncertainty of the mathematical model as well as the errors in the numerical objective evaluation caused, that not only the global minimizers, but also the local ones with the sufficiently low objective value may represent the solutions. A large variety of such problems in mechanics and other branches may be found in the literature. The good example was published by Cabib, Davini and Chong-Quing Ru [3] where the ambiguity of the global minimizer was mathematically proved.

The only way to solve such kind of problems is to find all local minimizers that satisfies the additional criterion (e.g. objective is lower then the assumed threshold). Niching genetic algorithms as well as multi start strategies were often applied as the robust methods for finding multiple minimizers (see e.g. [8], [9]). The adaptive strategies that can significantly decrease the number of objective calls are also highly appreciated (see e.g. [4]).

We propose the new strategy called *hp*–HGS composed of the adaptive numerical method *hp*–FEM (see Demkowicz [5], [6]) that allow very effective direct problem solving up to the assumed accuracy expressed by the energy error of the physical phenomenon and the multi-deme, Hierarchic Genetic Strategy HGS (see Schaefer, Kołodziej [10]) that allow economic global search with the adaptive accuracy in the parameters domain. The nature of HGS allow finding many local minimizers in parallel. The proper rule of scaling the error of solving direct problem vs. inverse problem error decreases the computational cost of the single objective call. The computational example show the *hp*–HGS behavior by solving the simple bimodal problem of heat conductivity identification. The parameter tunning as well as appearing of dangerous artifacts are discussed.

2 *hp*–HGS

The HGS introduced by Kołodziej and Schaefer (see e.g. [10], [11]) proceeds tree-structured, dynamically changing set of dependent demes. The depth of HGS tree is limited by $m < +\infty$. All demes work asynchronously and are synchronized by the message-passing mechanism if necessary. The evolution of each deme is governed by the separate instance of the Simple Genetic Algorithm (SGA) (see Vose [15]).

The low-order demes (closer to the root) perform more chaotic search with the lower accuracy, while the demes of higher order perform the more accurate, local search. The various search accuracy is obtained by the various encoding precision and by the different length binary strings as the genotypes in demes of different order. The unique deme of the first order (root) utilizes the shortest genotypes, while the leafs utilizes the longest ones. To obtain the search coherency for demes of different order the special kind of hierarchical, nested encoding is used. First

the densest mesh of phenotypes in $\mathcal{D}$ for the demes of m-th order is defined. Next the meshes for the lower order demes are recursively defined by selecting some nodes from the previous ones. The maximum diameter of the mesh δ_j associated with the demes of the order j determines the search accuracy at this level of the HGS tree. Of course $\delta_1 >, \ldots, > \delta_m$.

Each deme expecting leaf-demes sprouts the new child-deme after the constant number of genetic epochs K called the *metaepoch*. The child-deme is activated in the promising region of the evolutionary landscape surrounding the best fitted individual distinguished from the parental deme at the end of the metaepoch.

HGS implements also two mechanisms that allow to reduce the search redundancy. The first one called *conditional sprouting* disable to sprout new deme in the region already occupied or explored by the brother-deme (another child-deme of the same order sprouted by the same parent). The second mechanism called *branch reduction* reduces the branches of the same order that perform the search in the common landscape region or in the region already explored.

Let us apply HGS for solving the inverse problem (2). The fitness function for the particular deme should be based on the energy error

$$e_{h,p}(g) = \left| J_{h,p}(g) - J(\hat{d}) \right| \tag{3}$$

computed by using *hp*–FEM which approximate the objective function of the global optimization problem (2) for the particular values of h and p. As previously, $\hat{d} \in \mathcal{D}$ denotes the exact parameter value and $J(\hat{d})$ the known, exact energy of the exact solution while $J_{h,p}(g)$ the approximated value of energy computed by *hp*–FEM with respect to the parameter value $g \in \mathcal{D}$ obtained from the HGS individuals' genotype.

Let us assume for a while that g represents the parameter value decoded from the genotype that appears in the HGS deme of the j-th order, $j \in \{1, \ldots, m\}$. Using the Lemma 2 in [14] and the formula (8) in [12] that follow the regression of the error (3) while improving the *hp*–FEM approximation we obtain

$$e_{\frac{h}{2},p+1}(g) \leq \|u_{\frac{h}{2},p+1}(g) - u_{h,p}(g)\|_E^2 + \|u(g) - u_{h,p}(g)\|_E^2 + L|g - \hat{d}|. \tag{4}$$

The left-hand-side $e_{\frac{h}{2},p+1}(g) = \left| J_{\frac{h}{2},p+1}(g) - J(\hat{d}) \right|$ stands for the energy error while the first right-hand-side component $err_{FEM}(g) = \|u_{\frac{h}{2},p+1}(g) - u_{h,p}(g)\|_E^2$ is the relative error decrement in the single *hp*–FEM step (see Demkowicz [5]). Moreover L stands for the Lipshitz constants of the functional J and $|g - \hat{d}|$ is the error of the inverse problem solution that characterizes the individuals belonging to the HGS demes of the j-th order. It is easy to observe, that $|g - \hat{d}|$ corresponds to δ_j. The above formula shows, that the error of the energy evaluation over the fine FEM mesh is restricted by the relative FEM error on the coarse FEM mesh solution with respect to the fine mesh solution plus the absolute FEM error over the coarse FEM mesh plus the accuracy of the proper HGS branch.

The main idea of *hp*–HGS is to adjust dynamically the accuracy of the objective computation to the particular value of the parameter g encoded in the

```
1. if (j = 1) then
2.    initialize the root deme;
3. end if
4. t ← 0;
5. repeat
6.    if (global_stop_condition received) then
7.       STOP;
8.    end if
9.    for (i ∈ P^t) do
10.      solve the direct problem for g = code(i) on the coarse and fine FEM meshes;
11.      compute err_FEM(g);
12.      while (err_FEM(g) > Ratio * δ_j) do
13.         execute one step of hp adaptivity;
14.         solve the problem on the new coarse and fine FEM meshes;
15.         compute err_FEM(g);
16.      end while;
17.      compute fitness f_j(i) using the FEM mesh finally established;
18.   end for
19.   if (j > 1) then
20.      compute the phenotypes' average and send it to the parental deme;
21.      if (branch_stop_condition(P^t)) then
22.         STOP;
23.      end if
24.   end if
25.   if (((t mod K) = 0) ∧ (j < m)) then
26.      distinguish the best fitted individual x from deme P^t;
27.      if (¬ children_comparison(x)) then
28.         sprout;
29.      end if
30.   end if
31.   perform proportional selection, obtaining multiset of parents;
32.   perform SGA genetic operations on the multiset of parents;
33.   t ← t + 1;
34. until (false)
```

Algorithm 1. Pseudo-code of the j-th order deme P in the hp–HGS tree

individuals' genotype as well as for the inverse problem error that characterizes the current HGS branch. It may be obtained by balancing the components of the FEM error given by the right hand side of the formula (4), assuming δ_j as the accuracy of inverse problem solving by the branch of j-th order. We will perform then the hp-adaptation of the FEM solution of the direct problem while the quantity err_{FEM}/δ_j is greater then the assumed *Ratio*, which stands for the parameter of this strategy. The value of the parameter *Ratio* corresponds to the Lipshitz constant L, because two first components of the right-hand-side of (4) asymptotically vanish as a consequence of the hp–FEM convergence.

Notice, that no matter how the fitness of the individual i is computed by iterative process of the *hp*–FEM adaptation, the fitness function f_j is well defined for all individuals in branches of j-th order (it is not a random variable).

The draft of the single *hp*–HGS deme activity is stressed in the pseudo-code Algorithm 2. The function $branch\_stop\_condition(P)$ returns *true* if it detects the lack of evolution progress of the current deme P. The separate module continuously checks whether the satisfactory solution was found or *hp*–HGS could not find more local extremes. If yes, the *global_stop_condition* signal is send to all computing demes. The *conditional sprouting* mechanism is implemented as follows. Each branch excepting root computes the average of its phenotypes and send it to its parental deme. These values are analyzed by the $children\_comparison(x)$ procedure and compared with the phenotype of the best fitted individual x distinguished from the parental deme. This procedure returns *true* if x is sufficiently close to the existing child-demes. The *branch reduction* mechanism is omitted in Algorithm 2 for the sake of simplicity.

3 Sample Bimodal Inverse Problem

The inverse problem under consideration is originated from the classical L-shape domain direct problem, a model academic problem formulated in [1], [2], to test the convergence of the p and hp adaptive algorithms. The direct L-shape domain problem consists in computing the temperature distribution over the L-shape domain, presented in Figure 1 with fixed zero temperature in the internal part of the boundary, and the Neumann boundary condition prescribing the heat transfer on the external boundary. There is a single singularity in the central point of the domain, where the gradient of temperature $\|\nabla u\|$ goes to infinity, so the accurate numerical solution requires a sequence of adaptations in the direction of the central point.

The strong formulation of the direct problem consists in finding the temperature distribution $u \in C^2\left(\mathbb{R}^2\right)$ such that

$$\nabla \cdot K \nabla u = 0 \ \text{ in } \ \Omega \subset \mathbb{R}^2 \tag{5}$$

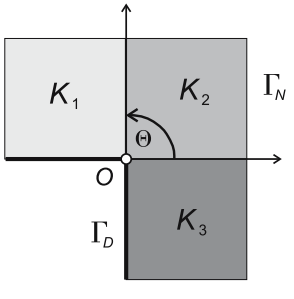


Fig. 1. The L-shape domain problem

with boundary conditions

$$u = 0 \text{ on } \Gamma_D, \quad \frac{\partial u}{\partial n} = \chi = r^{\frac{3}{2}} sin^{\frac{3}{2}} \left(\theta + \frac{\pi}{2}\right) \text{ on } \Gamma_N \tag{6}$$

with n being the unit normal outward to $\partial\Omega$ vector, and (r, θ) the radial system of coordinates with the origin point O (see Figure 1). K is the heat transfer coefficient, that takes the constant values K_1, K_2, K_3 over three parts of the domain, as illustrated in Figure 1.

The strong formulation (5) – (6) is transformed to the weak one: Find $u \in V = \{v \in L^2(\Omega); \int_\Omega \|v\|^2 + \|\nabla v\|^2 dx < +\infty, tr(v) = 0 \text{ on } \Gamma_D\}$ such that:

$$b(d; u, v) = l(v) \;\; \forall v \in V; \;\; b(d; u, v) = \int_\Omega K\, uv\, dx, \;\; l(v) = \int_{\Gamma_N} \chi\, v\, dS \tag{7}$$

where $d = (K_1, K_2, K_3) \in \mathcal{D} = [0, 3]^3 \subset \mathbb{R}^3$ is the vector of parameters.

The benchmark inverse problem is formulated in the following way: Find values of the heat transfer coefficients $\hat{g} \in \mathcal{D}$ that satisfy (2). The energy of the "exact" solution, was computed as $J_{h,p}(\hat{d})$, where $\hat{d} = (0.1, 1.5, 2.9)$ are the assumed values of the heat transfer coefficients. In order to obtain the energy $J_{h,p}(\hat{d}) = \int_\Omega \|u\|^2 + \|\nabla u\|^2\, dx$ close to the $J(\hat{d})$ the L-shape domain problem was computed once with the very high relative accuracy 10^{-5} by using the self-adaptive *hp*-FEM code.

Because both boundary conditions are symmetric with respect to the domain geometry (e.g. with respect to the axis $\theta = \pi/4$) then the parameter vector $\hat{d}' = (2.9, 1.5, 0.1)$ gives the same energy as $\hat{d}$, so the inverse problem (2) has two solutions in this case.

4 Test Results

We applied the three level's *hp*–HGS algorithm for solving inverse problem described in previous section. Because of the high computational cost of the accurate fitness evaluation, we decide to set *Ratio* to 900. Each metaepoch contains only three epochs. The next step of tunning the algorithm was to choose the mutation and crossing rates on each level. In order to do it properly we analyze the standard deviation of phenotypes for different values of the binary mutation rate (see Table 1). The mutation rate should decrease from the root to the leafs in order to increase the search locality in higher tree levels. From the other side, the standard deviation of the phenotypes obtained by the mutation has to be greater then the search accuracy δ_j on each level j in the *hp*–HGS tree in order to assure the enough search efficiency. The first choice was 0.3 for the crossing rate on each level of *hp*–HGS tree, and the mutation rates 0.05, 0.035, 0.005 for the root, the branches and leafs respectively. Many tests show that such setting allow to find only one solution to the inverse problem, mostly because of the small diversity of the root and the branches. The corrected setting of the mutation and crossing rates as well as other *hp*–HGS parameters are contained in Table 2. We pre-define the number of meataepoch as the stopping condition of our algorithm.

Table 1. Standard deviations of phenotypes for various mutation rates tested for all levels of the *hp*–HGS tree

Mutation rate	level 1	leve 2	level 3
0.5	1.54	1.51	1.51
0.2	1.29	1.19	1.21
0.1	0.85	0.91	0.83
0.035	0.49	0.55	0.56
0.025	0.4	0.53	0.55
0.01	0.31	0.23	0.25
0.005	0.24	0.18	0.11

Table 2. Final parameters of the *hp*–HGS tree applied by solving the sample inverse problem

	level 1	level 2	level 3
Code length	9	18	27
δ_j	0.375	0.047	0.006
Population size	50	12	8
Crossing rate	0.7	0.5	0.2
Mutation rate	0.5	0.1	0.01

Table 3. Leaves statistics obtained for the 5 selected *hp*–HGS settings

	test 1	test 2	test 3	test 4	test 5
number of leafs	21	21	10	21	21
err_{fit}	1.29	3.67	0.74	0.29	0.71
min $eucl_1$	0.65	0.62	2.02	0.22	0.89
min $eucl_2$	1.42	0.93	0.21	1.29	0.68
aver err_{fit}	30,38	98,63	230,81	179,33	66,48
aver $eucl_1$	0,98	1,28	2,02	1,16	1,6
aver $eucl_2$	2,03	1,69	0,9	1,33	0,89

The Table 3 shows the result of five selected runs of *hp*–HGS algorithm with the final parameters. The min $eucl_1$ and min $eucl_2$ mean the smallest euclidean distance between the computed solutions and the first exact solution $\hat{d} = (0.1, 1.5, 2.9)$ and the second exact solution $\hat{d}' = (2.9, 1.5, 0.1)$ respectively. The values aver $eucl_1$, aver $eucl_2$ are the averages of the euclidean distance between the best solutions found by leafs, which are in the attractor of the proper exact solution.

Perhaps the best result gave tests number two and five. In these cases both extremes were found by the separate leafs. In the test number one there are also both solutions found, but the second with much worst accuracy. It is also visible in the average distance between computed and real solutions. The same situation appears in the test number four. Because of the shorter assumed computational period in the third test, there were only about the half leafs than in others runs. It would be a reason, why the algorithm detected only one solution in this case.

In the case of multimodal problems the culmination of individuals may by located not only in the basins of attractions of global or local extremes. It may appear out of them when two solutions are linked to each other through mountain ridges with shallow saddle and the expected distance of the offspring from the parent is above half of the distance between these solutions. Such artifact looking like the optimum found was described by Karcz-Dulęba [7] and called *evolutionary channel*. Such phenomenon appears in about 15% of our tests. In these tests at least two leafs found both exact solution with the accuracy comparable to those presented in tests number two and five. Moreover, the point laying in the half way between two solutions were encountered as the solution by the other single leaf. This was probably caused by the unfortunate setting of the mixing parameters that results in the standard deviation of the phenotypes 1.6, almost the half of the distance between exact solutions, which is equal to 3.96. The flat shape of the fitness between two solutions and the assumed (by setting large *Ratio*) moderate accuracy of the *hp*–FEM caused that the fitness value at the artifacts could not be used to discard them from the set of potential solutions. The only way to eliminate the artifacts is to take into account their rare appearance in comparison to the right solutions of the inverse problem.

5 Conclusions

- The main difficulties in solving inverse parametric problems are caused by their ill posedness, in particular by the solution ambiguity, the low stability (sensitivity for the numerical errors) and high computational cost of the objective evaluation. The sophisticated global optimization strategies can be applied in order to find multiple solutions, keeping the memory and computational costs at the acceptable level.
- The proposed *hp*–HGS strategy is well suited for solving ambiguous inverse problems, because of its global search possibilities (local, parallel search controlled by the deme hierarchy). It is confirmed by the test results for the simple bimodal benchmark. Its well asymptotic properties can be also verified mathematically (see [13]).

- The *hp*–HGS strategy offers two ways to decrease the computational and memory costs. Firstly it is obtained by decreasing of the number of the objective calls by using the adaptation of the inverse problem accuracy (HGS strategy). Secondly, the cost of the direct problem solution which is necessary for fitness evaluation is decreased by the proper scaling of the FEM error using the *hp* adaptation technique.
- The parameters of *hp*–HGS have to be carefully tunned. The genetic mixing (mutation with crossover) has to be intensive enough in the higher levels in order to ensure the sufficient diversity of the root and high order branches. The standard deviation of mixing in leafs has to be correlated with the maximum search accuracy. Moreover, it needs to be "desynchronized" with the fitness periodicity (it should be significantly less then the half of the distance between each pair of solutions) in order to avoid artifacts that may be recognized as a solutions.
- No matter how SGA with the binary encoding is sometimes criticized as a tool for solving optimization problems in the continuous domains, it is used in each branch of the first version of *hp*–HGS mainly because the theoretical results that characterize its asymptotic behavior are available. We plan to design the next version of *hp*–HGS which will be based on the hierarchic genetic strategy with the real number encoding (see [11], [16]). Moreover the much broader testing program will be performed in order to evaluate the *hp*–HGS particular behaviors statistics and in order to confirm its advantages in solving inverse problems of the considered type.

Acknowledgment

The work has been partially supported by the Polish Ministry of Scientific Research and Information Technology and by the Foundation for Polish Science under Homming Program.

References

1. Babuška, I., Guo, B.: The *hp*-version of the finite element method, Part I: The basic approximation results. Comput. Mech. 1, 21–41 (1986)
2. Babuška, I., Guo, B.: The *hp*-version of the finite element method, Part II: General results and applications. Comput. Mech. 1, 203–220 (1986)
3. Cabib, E., Davini, C., Ru, C.-Q.: A problem in the optimal design of networks under transverse loading. Quarterly of Appl. Math. XLVIII(2), 251–263 (1190)
4. Cabib, E., Schaefer, R., Telega, H.: A Parallel Genetic Clustering for Inverse Problems. In: Kågström, B., Elmroth, E., Waśniewski, J., Dongarra, J. (eds.) PARA 1998. LNCS, vol. 1541, pp. 551–556. Springer, Heidelberg (1998)
5. Demkowicz, L.: Computing with *hp*-Adaptive Finite Elements. In: Applied Mathematics and Nonlinear Science. Chapmann & Hall / CRC, Boca Raton (2006)
6. Demkowicz, L., Kurtz, J., Pardo, P., Paszyński, M., Rachowicz, W., Zdunek, A.: Computing with hp-Adaptive Finite Elements. In: Applied Mathematics and Nonlinear Science. Frontiers: Three-Dimensional Elliptic and Maxwell Problems with Applications, vol. II. Chapmann & Hall /CRC, Boca Raton (2007)

7. Galar, R., Karcz-Dulęba, I.: The Evolution of Two. An Example of Space of States Approach. In: Sebald, A.V., Fogel, L.J. (eds.) Proc. of the Third Annual Conf. on Evolutionary Programming, San Diego, CA, pp. 261–203. Word Scientific, Singapore (1994)
8. Koper, K.D., Wysession, M.E., Wiens, D.A.: Multimodal function optimization with a niching genetic algorithm: A seismological example. Bulletin of the Seismological Society of America 89(4), 978–988 (1999)
9. Liszkai, T.R., Raich, A.M.: Solving Inverse Problems in Structural Damage Identification Using Advanced Genetic Algorithm Representation. In: Herkovits, J., Mazorche, S., Canelas, A. (eds.) Proc. of the 6th Congress of Structural and Multidisciplinary Optimization. ISSMO 2005, Rio de Janeiro, Brazil, May 30- June 3 (2005)
10. Schaefer, R., Kołodziej, J.: Genetic search reinforced by the population hierarchy. In: De Jong, K.A., Poli, R., Rowe, J.E. (eds.) Foundations of Genetic Algorithms 7, pp. 383–399. Morgan Kaufman Publisher, San Francisco (2003)
11. Schaefer, R.: Foundation of Genetic Global Optimization. Studies in Computational Intelligence Series, vol. 74. Springer, Heidelberg (2007) (with the chapter 6 written by Telega, H.)
12. Schaefer, R., Barabasz, B., Paszńyski, M.: Twin adaptive scheme for solving inverse problems. In: Proc. of the 10th Conf. on Evolutionary Algorithms and Global Optimization KAEiOG 2007, Będlewo, pp. 241–249 (2007)
13. Schaefer, R., Barabasz, B.: Asymptotic behavior of hp-HGS (hp-adaptive Finite Element Method coupled with the Hierarchic Genetic Strategy) by solving inverse problems. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part III. LNCS, vol. 5103, pp. 682–691. Springer, Heidelberg (2008)
14. Paszyński, M., Barabasz, B., Schaefer, R.: Efficient adaptive strategy for solving inverse problems. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 342–349. Springer, Heidelberg (2007)
15. Vose, M.D.: The Simple Genetic Algorithm. MIT Press, Cambridge (1999)
16. Wierzba, B., Semczuk, A., Kołodziej, J., Schaefer, R.: Hierarchical Genetic Strategy with real number encoding. In: Proc. of the 6th Conf. on Evolutionary Algorithms and Global Optimization Łagów Lubuski 2003, pp. 231–237. Warsaw Technical University Press (2003)

Author Index