

Lecture Notes in Computer Science

2809

Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

Springer

Berlin

Heidelberg

New York

Hong Kong

London

Milan

Paris

Tokyo

Roberto Moreno-Díaz Franz Pichler (Eds.)

Computer Aided Systems Theory – EUROCAST 2003

9th International Workshop on Computer Aided Systems Theory
Las Palmas de Gran Canaria, Spain, February 24-28, 2003
Revised Selected Papers



Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Roberto Moreno-Díaz
Universidad de Las Palmas de Gran Canaria
Instituto Universitario de Ciencias y Tecnologías Cibernéticas
Campus de Tafira, 35017, Las Palmas de Gran Canaria, Las Palmas, Spain
E-mail: rmoreno@ciber.ulpgc.es

Franz Pichler
Johannes Kepler University Linz
Institute of Systems Science
Altenbergerstr. 69, 4040 Linz, Austria
E-mail: pichler@cast.uni-linz.ac.at

Cataloging-in-Publication Data applied for

A catalog record for this book is available from the Library of Congress

Bibliographic information published by Die Deutsche Bibliothek
Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie;
detailed bibliographic data is available in the Internet at <<http://dnb.ddb.de>>.

CR Subject Classification (1998): J.6, I.6, I.2, J.7, J.3, C.1.m, F.4, F.3

ISSN 0302-9743

ISBN 3-540-20221-8 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

www.springeronline.com

© Springer-Verlag Berlin Heidelberg 2003
Printed in Germany

Typesetting: Camera-ready by author, data conversion by PTP-Berlin GmbH
Printed on acid-free paper SPIN 10949948 06/3142 5 4 3 2 1 0

Preface

The concept of CAST as Computer Aided Systems Theory, was introduced by F. Pichler of Linz in the late 80's to include those computer theoretical and practical developments as tools to solve problems in System Science. It was considered as the third component (the other two being CAD and CAM) that will provide for a complete picture of the path from Computer and Systems Sciences to practical developments in Science and Engineering.

The University of Linz organized the first CAST workshop in April 1988, which demonstrated the acceptance of the concepts by the scientific and technical community. Next, the University of Las Palmas de Gran Canaria joined the University of Linz to organize the first international meeting on CAST, (Las Palmas February 1989), under the name EUROCAST'89, that was a very successful gathering of systems theorists, computer scientists and engineers from most of European countries, North America and Japan.

It was agreed that EUROCAST international conferences would be organized every two years. Thus, the following EUROCAST meetings took place in Krems (1991), Las Palmas (1993), Innsbruck (1995), Las Palmas (1997), Vienna (1999) and Las Palmas(2001), in addition to an extra-European CAST Conference in Ottawa in 1994. Selected papers from those meetings were published by Springer-Verlag Lecture Notes in Computer Science nos. 410, 585, 763, 1030, 1333, 1728 and 2178 and in several special issues of Cybernetics and Systems: an International Journal. EUROCAST and CAST meetings are definitely consolidated, as it is demonstrated by the number and quality of the contributions over the years.

EUROCAST 2003 (Las Palmas, February 2003) continued with new approach to the Conferences which was adopted in 2001. Besides the classical core on generic CAST, there were other specialized workshops devoted to Complex Systems (chaired by Pichler and Moreno-Díaz), Neuroimaging and Neuroinformatics, (chaired by Alzola and Westin), Computational Methods in Biomathematics (chaired by Ricciardi), Natural and Artificial Neural Nets (chaired by Mira and Moreno-Díaz), Distributed Computing (chaired by Freire) and a Special Session on Autonomous Systems, promoted by García de la Rosa and Maravall.

This volume contains sixty selected full papers from the oral presentations of the different Sessions. The editors would like to thank all contributors for their quickness in providing their material in hard and electronic forms. Special thanks are due to Dr. Alexis Quesada, from the Institute of Cybernetics of the University of Las Palmas, for his great help in the preparation of the volume and to the Staff of Springer-Verlag Heidelberg for their valuable support.

Table of Contents

Complex Systems Tools and Applications

On Modeling and Simulation of Flows of Water by 3D-Cellular Automata	1
<i>Franz Pichler</i>	
Representation and Processing of Complex Knowledge	10
<i>Rudolf F. Albrecht, Gábor Németh</i>	
How Many Rounds to KO?, or Complexity Increase by Cryptographic Map Iteration	19
<i>Juan David González Cobas, José Antonio López Brugos</i>	
A Non-standard Genetic Algorithm Approach to Solve Constrained School Timetabling Problems	26
<i>Cristina Fernández, Matilde Santos</i>	
Application of Uncertain Variables to Task and Resource Distribution in Complex Computer Systems	38
<i>Z. Bubnicki</i>	
A Framework for Modelling the User Interaction with a Complex System	50
<i>M^a L. Rodríguez Almendros, M^a J. Rodríguez Fórtiz, M. Gea Megías</i>	
A Categorical Approach to NP-Hard Optimization Problems	62
<i>Liara Aparecida dos Santos Leal, Dalcidio Moraes Claudio, Laira Vieira Toscani, Paulo Blauth Menezes</i>	

Logic and Formal Tools

A Formulation for Language Independent Prelogical Deductive Inference	74
<i>Josep Miró</i>	
Multi-agent Simulation in Random Game Generator	83
<i>Takuhei Shimogawa</i>	
The Zero Array: A Twilight Zone	92
<i>Margaret Miró-Julíá</i>	
Invariants and Symmetries among Adaptive Agents	104
<i>Germano Resconi</i>	

Generalizing Programs via Subsumption	115
<i>Miguel A. Gutiérrez-Naranjo, José A. Alonso-Jiménez,</i>	
<i>Joaquín Borrego-Díaz</i>	

Social and Intelligent Systems

Modeling with Archetypes: An Effective Approach to Dealing with Complexity	127
<i>Markus Schwaninger</i>	

Equal Opportunities Analysis in the University: The Gender Perspective	139
<i>I.J. Benítez, P. Albertos, E. Barberá, J.L. Díez, M. Sarrió</i>	

Approximate Solutions to Semi Markov Decision Processes through Markov Chain Montecarlo Methods	151
<i>Arminda Moreno-Díaz, Miguel A. Virto, Jacinto Martín,</i>	
<i>David Ríos Insua</i>	

Knowledge Base for Evidence Based Medicine with Bioinformatics Components	163
<i>Witold Jacak, Karin Pröll, Jerzy Rozenblit</i>	

Diversified Approach to Methodology and Technology in Distributed Intelligent Building Systems	174
<i>Andrzej Jablonski, Ryszard Klempous, Benedykt Licznarski</i>	

Temporal Approaches in Data Mining. A Case Study in Agricultural Environment	185
<i>Francisco Guil, Alfonso Bosch, Samuel Túnez, Roque Marín</i>	

Personalized Guided Routes in an Adaptive Evolutionary Hypermedia System	196
<i>Nuria Medina-Medina, Fernando Molina-Ortiz, Lina García-Cabrera,</i>	
<i>José Parets-Llorca</i>	

Temporal Data Management and Knowledge Acquisition Issues in Medical Decision Support Systems	208
<i>M. Campos, J. Palma, B. Llamas, A. González, M. Menárguez,</i>	
<i>R. Marín</i>	

Distributed Computing

Development of a Scalable, Fault Tolerant, and Low Cost Cluster-Based e-Payment System with a Distributed Functional Kernel	220
<i>C. Abalde, V. Gulías, J. Freire, J. Sánchez, J. García-Tizón</i>	

Generative Communication with Semantic Matching in Distributed Heterogeneous Environments	231
<i>Pedro Álvarez, José A. Bañares, Eloy J. Mata, Pedro R. Muro-Medrano, Julio Rubio</i>	
Mapping Nautilus Language into Java: Towards a Specification and Programming Environment for Distributed Systems	243
<i>Claudio Naoto Fuzitaki, Paulo Blauth Menezes, Júlio Pereira Machado, Simone André da Costa</i>	
Design of a Medical Application Using XML Based Data Interchange	253
<i>C. Mariño, C. Abalde, M.G. Penedo, M. Penas</i>	
Partial-Order Reduction in Model Checking Object-Oriented Petri Nets	265
<i>Milan Češka, Luděk Haša, Tomáš Vojnar</i>	
On the Strong Co-induction in Coq	279
<i>J.L. Freire Nistal, A. Blanco Ferro, Victor M. Gulías, E. Freire Brañas</i>	

Autonomous and Control Systems

A Throttle and Brake Fuzzy Controller: Towards the Automatic Car	291
<i>J.E. Naranjo, J. Reviejo, C. González, R. García, T. de Pedro</i>	
ADVOCATE II: ADVanced On-Board Diagnosis and Control of Autonomous Systems II	302
<i>Miguel Angel Sotelo, Luis Miguel Bergasa, Ramón Flores, Manuel Ocaña, Marie-Hélène Doussin, Luis Magdalena, Joerg Kalwa, Anders L. Madsen, Michel Perrier, Damien Roland, Pietro Corigliano</i>	
Segmentation of Traffic Images for Automatic Car Driving	314
<i>Miguel Ángel Patricio, Darío Maravall</i>	
Vision Based Intelligent System for Autonomous and Assisted Downtown Driving	326
<i>Miguel Ángel Sotelo, Miguel Ángel García, Ramón Flores</i>	
Using Fractional Calculus for Lateral and Longitudinal Control of Autonomous Vehicles	337
<i>J.I. Suárez, B.M. Vinagre, A.J. Calderón, C.A. Monje, Y.Q. Chen</i>	

Computational Methods in Biomathematics

Recent Advances in the Walking Tree Method for Biological Sequence Alignment	349
<i>Paul Cull, Tai Hsu</i>	

Towards Some Computational Problems Arising in Biological Modeling	360
<i>Virginia Giorno, Amelia G. Nobile, Enrica Pirozzi, Luigi M. Ricciardi</i>	
Single Point Algorithms in Genetic Linkage Analysis	372
<i>Daniel Gudbjartsson, Jens A. Hansen, Anna Ingólfssdóttir, Jacob Johnsen, John Knudsen</i>	
A Self-adaptive Model for Selective Pressure Handling within the Theory of Genetic Algorithms	384
<i>Michael Affenzeller, Stefan Wagner</i>	
Computational Methods for the Evaluation of Neuron's Firing Densities	394
<i>Elvira Di Nardo, Amelia G. Nobile, Enrica Pirozzi, Luigi M. Ricciardi</i>	
Developing the Use of Process Algebra in the Derivation and Analysis of Mathematical Models of Infectious Disease	404
<i>R. Norman, C. Shankland</i>	
On Representing Biological Systems through Multiset Rewriting	415
<i>S. Bistarelli, I. Cervesato, G. Lenzini, R. Marangoni, F. Martinelli</i>	
Natural and Artificial Neural Nets	
A Model of Neural Inspiration for Local Accumulative Computation	427
<i>José Mira, Miguel A. Fernández, Maria T. López, Ana E. Delgado, Antonio Fernández-Caballero</i>	
Emergent Reasoning from Coordination of Perception and Action: An Example Taken from Robotics	436
<i>Darío Maravall, Javier de Lope</i>	
Inverse Kinematics for Humanoid Robots Using Artificial Neural Networks	448
<i>Javier de Lope, Rafaela González-Careaga, Telmo Zarraonandia, Darío Maravall</i>	
Neurosymbolic Integration: The Knowledge Level Approach	460
<i>J. Mira, A.E. Delgado, M.J. Taboada</i>	
On Parallel Channel Modeling of Retinal Processes	471
<i>J.C. Rodríguez-Rodríguez, A. Quesada-Arencibia, R. Moreno-Díaz jr., K.N. Leibovic</i>	

Geometric Image of Statistical Learning (Morphogenetic Neuron)	482
<i>Elisa Alghisi Manganello, Germano Resconi</i>	
Systems and Computational Tools for Neuronal Retinal Models	494
<i>Roberto Moreno-Díaz, Gabriel de Blasio</i>	

Neuroinformatics and Neuroimaging

A Novel Gauss-Markov Random Field Approach for Regularization of Diffusion Tensor Maps	506
<i>Marcos Martín-Fernández, Raul San Josá-Estépar, Carl-Fredrik Westin, Carlos Alberola-López</i>	
Coloring of DT-MRI Fiber Traces Using Laplacian Eigenmaps	518
<i>Anders Brun, Hae-Jeong Park, Hans Knutsson, Carl-Fredrik Westin</i>	
DT-MRI Images : Estimation, Regularization, and Application	530
<i>D. Tschumperlé, R. Deriche</i>	
An Efficient Algorithm for Multiple Sclerosis Lesion Segmentation from Brain MRI	542
<i>Rubén Cárdenes, Simon K. Warfield, Elsa M. Macías, Jose Aurelio Santana, Juan Ruiz-Alzola</i>	
Dynamical Components Analysis of FMRI Data: A Second Order Solution	552
<i>Bertrand Thirion, Olivier Faugeras</i>	
Tensor Field Regularization Using Normalized Convolution	564
<i>Carl-Fredrik Westin, Hans Knutsson</i>	
Volumetric Texture Description and Discriminant Feature Selection for MRI	573
<i>Abhir Bhalerao, Constantino Carlos Reyes-Aldasoro</i>	
White Matter Mapping in DT-MRI Using Geometric Flows	585
<i>Lisa Jonasson, Patric Hagmann, Xavier Bresson, Reto Meuli, Olivier Cuisenaire, Jean-Philippe Thiran</i>	
Anisotropic Regularization of Posterior Probability Maps Using Vector Space Projections. Application to MRI Segmentation	597
<i>M.A. Rodriguez-Florido, R. Cárdenes, C.-F. Westin, C. Alberola, J. Ruiz-Alzola</i>	
Fast Entropy-Based Nonrigid Registration	607
<i>Eduardo Suárez, Jose A. Santana, Eduardo Rovaris, Carl-Fredrik Westin, Juan Ruiz-Alzola</i>	

Image Processing

3D Reconstruction from a Vascular Tree Model 616
*Luis Álvarez, Karina Baños, Carmelo Cuenca, Julio Esclarín,
Javier Sánchez*

ESKMod, a CommonKADS Knowledge Model Integrating
Multiple Classic Edge Based Segmentation Algorithms 627
Isabel M. Flores-Parra, J. Fernando Bienvenido

Frequency Analysis of Contour Orientation Functions for
Shape Representation and Motion Analysis 639
Miguel Alemán-Flores, Luis Álvarez-León, Roberto Moreno-Díaz jr.

Preprocessing Phase in the PIETSI Project (Prediction of
Time Evolution Images Using Intelligent Systems) 651
J.L. Crespo, P. Bernardos, M.E. Zorrilla, E. Mora

Devices to Preserve Watermark Security in Image Printing
and Scanning 660
Josef Scharinger

Author Index 671

On Modeling and Simulation of Flows of Water by 3D-Cellular Automata

Franz Pichler

Johannes Kepler University Linz
Institute of Systems Science, Linz, Austria
pichler@cast.uni-linz.ac.at

1 Introduction

By an earlier paper [1] a 3-D cellular automata model CA was suggested to investigate a possible existence of stable clusters in liquid water in (or after) a state of turbulence. In order to take in the model molecular forces into account, we proposed a model for the molecular level of description. To make such a cellular model as simple as possible the geometrical form of each cell was assumed to be a (regular) cube. Regarding the topology of the cellular automata model CA we assumed for each cell c the associated neighborhood $N(c)$ consisting of the 6 cells $c_o, c_p, \dots, c_s$ which are adjacent to c ("von Neumann neighborhood").

Motivated by the electrostatic polarity of a H_2O molecule, the states (of a H_2O molecule) in a cell c have been defined by the $24=8 \times 3$ different possible geometrical positions x which a regular triangle (describing the geometrical form of H_2O molecule) can have, with the O-atom at one of the vertices and the two H-atoms being positioned on the two opposite vertices. Figure 1 shows by a graphic representation the 24 different geometrical positions x of a state (H_2O molecule) in a cell.

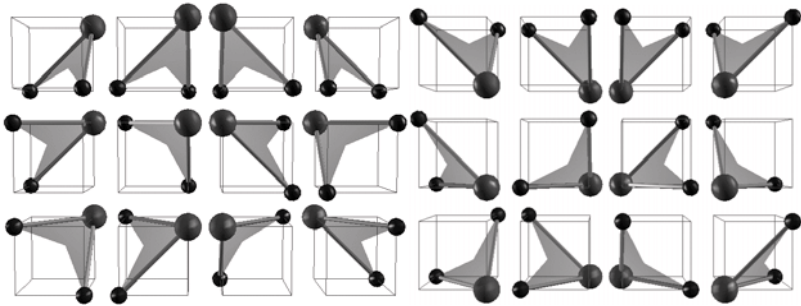


Fig. 1. Geometrical positions x of H_2O molecules in a cubic cell c

In addition to the geometrical positions x of states we assume also an associated directional moment y described by a binary vector $y=(y_o, y_p, \dots, y_s)$, where $y_i \in \{0,1\}$ for $i=0,1,\dots,5$. Each component y_i of y is interpreted as a vector of length 0 or 1 which is perpendicular to the corresponding i 'th plane of the cube.

In our paper [1] only the basic framework to construct the cellular automata CA by its individual cells c was given. In this paper we will present a detailed mathematical

construction of c and give also the necessary arguments such that the simulation model SIMCA for it can be implemented.

2 Construction of the Cell Automata C

We model the set X consisting of the 24 different positions of a H_2O molecule by $X := GF(8) \times GF(3)$. For $x = (x_p, x_2) \in X$ the value $x_p \in GF(8)$ determines the position of the O-atom at the 8 different possible vertices of the cube; the value $x_2 \in GF(3)$ of x gives the location of the corresponding two H-atoms. At this stage of modeling, the algebraic structure of the finite fields $GF(8)$ and $GF(3)$ play no role. This is reserved for a possible use of it at a later stage of modeling. By Y we denote the set $Y := GF(2)^6$ of directional moments. With this notation a state q of a H_2O molecule in the cell c can be described by $q = (x, y) \in X \times Y$. However, we have to consider in c also the situation, that no H_2O molecule is in the cell c . This is described by the symbol $\emptyset$, which we call the empty state. The state set Q of our cell automata c is therefore defined by $Q := (X \times Y) \cup \{\emptyset\}$ or also

$$Q := (GF(8) \times GF(3) \times GF(2)^6) \cup \{\emptyset\}.$$

The next point in our modeling task is to construct in a proper way the state transition function δ for the cell automata c of CA. By our assumption of a “von Neumann neighborhood” of c the function δ has to have the form $\delta: Q \times Q_0 \times Q_1 \times \dots \times Q_5 \rightarrow Q$ where Q_i ($i=0,1,2,\dots,5$) denotes the state sets of the neighboring cells $c_0, c_1, c_2, \dots, c_5$ respectively. The definition of the assignment of the next state $q^* = \delta(q, q_0, q_1, \dots, q_5)$ of c at time $t+1$ following the current state q under the influence of the neighbor states $q_0, q_1, \dots, q_5$ at time t is the crucial part of our modeling task. In a first step we decompose δ into $\delta = \delta_p \circ \delta_c$, where δ_p describes the propagation of states (H_2O molecules) between different cells and δ_c models the effect of collision of states (H_2O molecules) which are caused by its propagation.

In our model of a CA propagation of states can only happen in connection with empty states $\emptyset$. Therefore it is in our case sufficient to define δ_p only for the values $\bar{q} = \delta_p(\emptyset, q_0, q_1, \dots, q_5)$. The principal idea is that $\bar{q}$ is determined by a selection function α to become one of the states $q_0, q_1, \dots, q_5$. To approach a proper mathematical definition we need a few preliminary considerations.

- (1) For the set $Y = GF(2)^6$ of directional moments we define the function $\gamma: GF(2)^6 \rightarrow GF(6)$ which determines by its values $\gamma(y)$ the (preferred) direction which a H_2O molecule which is in state $q = (x, y)$ will take for propagation.

The determination of γ respects additional assumed constraints on the movement of H_2O molecules in CA which are caused by physical forces, e.g. by gravitational or shearing forces.

The value $\gamma(y)$ refers to the direction which is given by the corresponding plane of the cube through which the propagation preferable will take place.

- (2) By *select* we introduce a function which selects from the neighborhood $N(c)$ of a given cell c the subset $\bar{N}(c)$ of $N(c)$ which consists of all cells $\bar{c}$ with state $\bar{q} = (\bar{x}, \bar{y})$ where $\gamma(\bar{y})$ determines a direction “towards” c .

- (3) For the neighborhood $N(c)$ of a cell c we define the following ordering relation $\leq$
- (a) for $N(c) = \{c_0, c_1, c_2, c_3, c_4, c_5\}$ c_0 is minimal and c_5 is maximal with respect to $\leq$
 - (b) the ordering $\leq$ of the set $\{c_1, c_2, c_3, c_4\}$ is induced by the distances $d(c_i)$ of the cells c_i ($i=1,2,3,4$) with respect to the boundary of CA
- (4) By $\max: P(N(c)) \rightarrow P(N(c))$ we introduce the function given by $\max(U) := \{\bar{c}; \bar{c} \in U \ \& \ \forall c \in U \Rightarrow c \leq \bar{c}\}$.

The function $\max$ determines from any subset U of $N(c)$ the cells c of $N(c)$ which are maximal with respect to $\leq$.

After the introduction of (1)-(4) we are in the position to define the propagation $\delta_p(\emptyset, q_0, q_1, \dots, q_5)$ of a state $\bar{q} \in N(c)$ to replace the empty state $\emptyset$ in cell c .

$\bar{q} = \delta_p(\emptyset, q_0, q_1, \dots, q_5)$ is defined by $\bar{q} \in \alpha(N(c))$ where α denotes the composition $\alpha = \text{select} \circ \max$. In case that $\text{card}(\alpha(N(c))) = 1$ we have a deterministic state propagation; in the case that $\text{card}(\alpha(N(c))) > 1$ we have a non-deterministic state propagation. In this case we propose a probabilistic state propagation, where all relevant states of the neighborhood of c propagates into c with equal probability. In the case that $\alpha(N(c)) = \emptyset$ (empty set) we determine $\bar{q} = q_5$. In order to regulate a possible conflict, this rule dominates the propagation of q_5 with respect to other neighbor cells of c_5 .

Next we have to determine the collision function δ_c of δ . In general δ_c has to model the effect of physical forces associated to the neighboring cells which influence the embedding of a H_2O molecule into the cell c after propagation. δ_c has to be defined as a function of the kind

$$\delta_c: GF(8) \times GF(3) \times GF(2)^6 \times ((GF(8) \times GF(3) \times GF(2)^6) \cup \{\emptyset\})^6 \rightarrow GF(8) \times GF(3) \times GF(2)^6.$$

Certainly such a function is very complex and difficult to define such that given physical conditions are met.

In principle one can consider the following approaches

- (a) exploration of symmetries which are suggested by physical facts leading to equivalence relations which allow the construction of a quotient of δ_c and such reduce its complexity
- (b) assumption of equivalences in the arguments of δ_c which are based on heuristics and which led to the use of projections of δ_c
- (c) heuristical assumptions of algebraic forms or of algorithms which generate δ_c
- (d) determination of a incompletely specified collision function δ_c with the goal of stepwise improvements and completion by simulation experiments

The approach which we will suggest is a mixture of the above. In detail the following assumption will be made by us:

- (1) we neglect the influence of the directional moments y of the states in the neighborhood $N(c)$ of the cell c

- (2) we consider any neighboring cell $\bar{c}$ from which the state $\bar{q}$ has propagated to the cell c of no influence on the collision function δ_c . Taking (1) and (2) into account, the collision function δ_c reduces to the form $\delta_c: GF(8) \times GF(3) \times GF(2)^6 \times ((GF(8) \times GF(3)) \cup \{\emptyset\})^5 \rightarrow GF(8) \times GF(3) \times GF(2)^6$
- (3) for the embedding of $\bar{q}$ into the cell c by δ_c the physical forces which are caused by hydrogen bonding are considered to be dominant. Therefore the geometrical positions x_i of the states of the relevant neighboring cells should have an important influence on the next state q^* which is computed by δ_c . We want to determine q^* such that a maximum number of hydrogen bondings is achieved by δ_c .
- (4) for the transition of the propagated state $\bar{q}$ into q^* by δ_c we allow only a limited number of (rotational) movements of the position $\bar{x}$ of $\bar{q}$ to reach the embedded position x^* of q^*
- (5) the directional moment y^* of the next state q^* determined by δ_c depends only on the directional moment $\bar{y}$ of the state $\bar{q}$ and the selected (rotational) movement of $\bar{x}$ to x^* according to (4).

On the basis of our assumptions (3)-(5) we are in the position to assume a parallel decomposition of δ_c into $\delta_c = \delta_{c_1} \times \delta_{c_2}$ with

$$\delta_{c_1} : GF(8) \times GF(3) \times ((GF(8) \times GF(3)) \cup \{\emptyset\})^5 \rightarrow GF(8) \times GF(3)$$

and

$$\delta_{c_2} : GF(2)^6 \rightarrow GF(2)^6$$

where δ_{c_1} determines the effect of collision for the geometrical positions and δ_{c_2} determines the effect of collision for the directional moments.

In conclusion, by the above arguments it should be possible to determine by exhaustive or heuristic search the collision function δ_c and therefore also the state transition function $\delta = \delta_p \circ \delta_c$ of the cell machine $c = (Q^6, Q, \delta)$.

The following detail on the kind of our definition of the state transition function δ of a cell c should be mentioned: The state transition δ is not purely local acting on c , it is, as we have seen also changing the state of the neighboring cell $\bar{c}$ which is determined by α . This state goes by δ from $\bar{q}$ to $\emptyset$. However this “side effect” of δ is local and does not effect the dynamical behavior of the cellular automata CA in an unwanted manner. To the contrary, the dynamic movement of the empty states through the CA over time contributes to a visualization of the state process.

3 Construction of a 2-D Example for the Cell Automata C

To contribute to an easier understanding for our approach we want to give an example for the 2-D case. Certainly such a model has only little physical relevance since it

assumes a planar cellular automata which means in our context, the assumption of a water-container with a thickness of the size of a H_2O molecule.

For the geometrical form of a cell c we now assume a square which is surrounded by the 4 neighboring cells c_0, c_1, c_2, c_3 , also in form of squares, which constitute the “von Neumann neighborhood”. Figure 2 shows this.

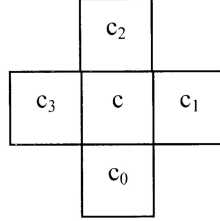
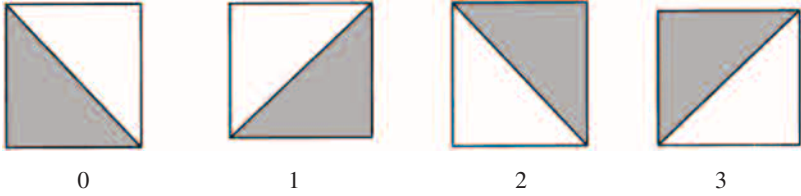


Fig. 2. Cell geometry of the 2-D example

Each cell automata c has a state set $Q=(X \times Y) \cup \{\emptyset\}$ which is now given by $X:=GF(3)$, which is the set of the 4 different possible positions of a H_2O molecule, and $Y:=GF(2)^4$, the set of the 16 different possible directional moments. Figure 3 shows the positions x of states and also the directional moments y .

(a) geometrical positions x



(b) directional moments y

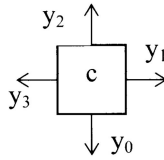


Fig. 3. Geometrical interpretation of the states $q=(x,y)$ in the 2-D example

For the construction of the propagation part δ_p of the state transition function we need to know the function $\gamma:GF(2)^4 \rightarrow GF(3)$ which assigns to each directional moment $y=(y_0, y_1, y_2, y_3)$ the direction $\gamma(y)$ to be associated to a state y of a cell c .

By reasonable physical arguments taking gravitation into account γ is given for our 2-D example by the following Table 1.

Table 1. Function $\gamma: GF(2)^4 \rightarrow GF(3)$ in the 2-D example

y_0	y_1	y_2	y_3	$\gamma(y)$
0	0	0	0	0
0	0	0	1	3
0	0	1	0	2
0	0	1	1	3
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	2
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	3
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

A necessary modification of γ to meet possible physical requirements better, should be based on simulation experiments.

For the order relation $\leq$ on $N(c) = \{c_o, c_p, c_{2p}, c_3\}$ we have, depending on the location of c in CA, the following cases

1. $c_o \leq c_1 \leq c_2$ and $c_o \leq c_3 \leq c_2$
2. $c_o \leq c_3 \leq c_1 \leq c_2$
3. $c_o \leq c_1 \leq c_3 \leq c_2$

With the determination of γ and $\leq$ the function $\alpha = select \circ max$ can be constructed and in consequence also δ_p .

The collision function δ_c has in the 2-D example of a CA in general the form $\delta_c: GF(3) \times GF(2)^4 \times ((GF(3) \times (GF(2)^4) \cup \{\emptyset\})^4 \rightarrow GF(3) \times GF(2)^4$.

By the reduction of δ_c and decomposition of δ_c into $\delta_c = \delta_{c_1} \circ \delta_{c_2}$ as proposed by us we get

$$\delta_{c_1}: GF(3) \times (GF(3) \cup \{\emptyset\})^3 \rightarrow GF(3)$$

and

$$\delta_{c_2}: GF(2)^4 \rightarrow GF(2)^4$$

The final determination of δ_{c_1} has to be done by a search procedure such that an optimal fitting of the state position x^* computed by δ_{c_1} is reached. For the search we limit the positional movements from $\bar{x}$ to x^* to rotational turning of a H_2O molecule with the angels $(.)^\circ$ given by $0^\circ, \pm 90^\circ$. In our 2-D case maximal 3 hydrogen bondings can be realized. The priority of found solutions for δ_{c_1} is given by the following Table 2.

Table 2. Priority of rotational movements

$(.)^\circ$	# bondings
0°	3
$\pm 90^\circ$	3
0°	2
$\pm 90^\circ$	2
0°	1
$\pm 90^\circ$	1

The function $\delta_{c_2} : GF(2)^4 \rightarrow GF(2)^4$ is determined in our example by the rotational movement of the positions found for δ_{c_1} .

$\delta_{c_2}(\bar{y}_0, \bar{y}_1, \bar{y}_2, \bar{y}_3) = (y_0^*, y_1^*, y_2^*, y_3^*)$ can be determined by the following Table 3, which describes δ_{c_2} by the appropriate permutation of the coordinates.

Table 3. The function δ_{c_2} in the 2-D example

$(.)^\circ$	y_0^*	y_1^*	y_2^*	y_3^*
0°	$\bar{y}_0$	$\bar{y}_1$	$\bar{y}_2$	$\bar{y}_3$
$+90^\circ$	$\bar{y}_1$	$\bar{y}_2$	$\bar{y}_3$	$\bar{y}_0$
-90°	$\bar{y}_3$	$\bar{y}_0$	$\bar{y}_1$	$\bar{y}_2$

In the case that no hydrogen bonding are found we take $q^* = \bar{q}$ (no effect of collision).

4 Further Modeling Issues

In our discussion we have so far covered the principal concepts for the construction of the cell automata $c=(Q^\delta, Q, \delta)$ of our cellular automata CA. For the case of a 2-D CA we have elaborated this discussion in greater detail. In this chapter we discuss the geometrical form which we suggest to build a simulation model for the 3-D CA and we discuss also the dynamization of our generative model. A first step in this discussion was already presented in a former paper [1].

In reality we think to model the flow of water in a conical container CT of the geometrical form shown in Figure 4.

In a real experiment we assume that water can be filled into CT at the top and that it flows out at the bottom. As we experience it from our observations at an usual sink in a bath-room this will in many cases cause a turbulent flow. The turbulence can partly be assumed to result from certain boundary conditions on the wall of the container.

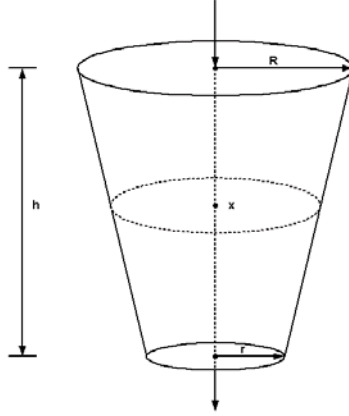


Fig. 4. Conical container CT

Our ultimate goal is to model this flow by the dynamics of the cellular automata CA and to investigate possible stable structures which might exist as part of (strange) attractors generated by chaos (turbulence). Such investigations can not be done by usual mathematical means and need a computational support by simulation. Cellular automata have proven in the past, that they provide powerful means for different modeling tasks in hydromechanics (see for example [2], [3], [4], [5]).

Our modeling approach differs however, in certain details from known research in that field. It is therefore, at this moment, still uncertain if our construction of a cellular automata for modeling liquid water, will show the expected dynamical behavior. Therefore, the improvement and tuning of our modeling approach has to depend on simulation experiments.

Of crucial importance is the kind of boundary conditions we assume for the CA. They are reflected in a specific operation of the propagation function δ_p of the state transition function δ for the cells c at the boundary.

A first approach is to determine for the boundary cells of the cone wall the function γ such that the direction $\gamma(y)$ of a state $q=(x,y)$ points always toward to the inners of the cone. This would model, we believe, to a certain extent the existence of shearing forces which contribute to a reflection of the H_2O molecule from the wall of the container.

For the cells on the top of the conical container CT it is assumed that at an empty state $\emptyset$ always will be supplied by a state $\bar{q} = (\bar{x}, \bar{y})$ (a H_2O molecule) from the environment on top of CT. The states $\bar{q}$ are randomly chosen, however $\bar{y}$ such that $\gamma(\bar{y}) = 0$ (pointing to the bottom). The cells at the bottom of the container CT, on the other hand, are considered to be permanently in empty state $\emptyset$. All states (H_2O molecules) which are propagated to such cells are, so to say, absorbed and contribute to the output of the CA.

By the construction of the cell machines $c=(Q^{\delta}, Q, \delta)$ parallel execution of the CA is assured and no conflicts concerning propagation of states can appear.

After initialization of the CA by a random choice for the different states in the cell machines and providing the required boundary conditions for the input and the output

of the CA it is expected that the flow of states (H_2O molecules) will stabilize at a certain equilibrium for the throughput.

5 Cellular Water Simulator SIMCA

To evaluate our model concerning the dynamics of the flow of states (H_2O molecules) through the container CT a simulator for cellular automata (SIMCA) has been implemented and first experiments have been performed [6]. SIMCA allows stepwise execution of the model as well as also to run state trajectories, both supported by visualization.

A goal of our future work is to perform experiments by tuning the model parameters and, if necessary, by modifying the state transition function of the cells until expected stable structures which show the effect of hydrogen bondings between the states (H_2O molecules) are discovered.

It is important that after this goal is reached a validation of the model with respect to physical facts, which have to be respected, is done. In this way we believe we are able to contribute in the discussion to the topic “water has memory” which is addressed in recent popular however scientifically often sloppy publications.

References

1. Pichler, F.: A basic framework for modeling the flow of water by cellular automata. BIOCAMP 2002, Vietri Sul Mare, Italy. Accepted for publication
2. Wolfram, Stephen: Theory and Applications of Cellular Automata. World Scientific Singapore (1986)
3. Wolfram, Stephen: Cellular Automata and Complexity. Westview Press (1994)
4. Chopard, Bastien; Droz, Michel: Cellular Automata Modeling of Physical Systems. Cambridge University Press, Cambridge (1988)
5. Rivet, Jean-Pierre; Boon, Jean-Pierre: Lattice Gas Hydrodynamics. Cambridge University Press, Cambridge (2001)
6. Fried, Alexander: Cellular Water Simulator. Technical Report (11 pages). Systems Theory and Information Technology, University Linz, March 2003

Representation and Processing of Complex Knowledge

Rudolf F. Albrecht¹ and Gábor Németh²

¹ Faculty of Natural Science, University of Innsbruck, 25, Technikerstr., 6020 Innsbruck, Austria

Rudolf.Albrecht@uibk.ac.at

² Department. of Telecommunication, Budapest University of Technology and Economics, 2, Magyar tudósok krt., 1117 Budapest, Hungary
nemeth@hit.bme.hu

Abstract. In this article results of earlier publications on the subject are summarized with some extensions and notational improvements. In the introduction a heuristic outline of human knowledge, its acquisition and its processing is given to motivate the following mathematical representation. The basic mathematical concepts are hierarchies of generalized relations and operations on these, corresponding to knowledge and knowledge processing. Included are objects with space and time parameterization, valuations of objects by partial ordered values, neighborhood and similarity relations, i.e. topologies, and variable objects.

1 Introduction

Based on our physical and mental capabilities, we are able to gain knowledge of what we consider to be "our real world" by memorizing perceptions and recognitions of real world objects and their properties in form of mental objects like impressions, conceptions, notions, structural and comparative relationships, causal dependencies, relationships with respect to our understanding of time and space.

By thinking according our laws of thinking, which is our mental knowledge processing, we are able to generate from present knowledge new mental objects, which may or may not correspond to perceptible objects of our real world. Thinking activities are for example hierarchical classifications, composition and decomposition of objects according rules, causal conclusions, recognition of equality, similarity, proximity, time dependent behavior, symbolization of objects, recognition of quantitative and qualitative properties. The totality of our mental objects may change by forgetting, further knowledge acquisition and the results of thinking processes.

For the representation of mental objects and their processing we use symbolizations in terms of classical mathematics. Defined and derived symbols are assumed to have an interpretation; several symbols may denote the same object. For example, the abstract notion "one" is symbolized by *one* or *1* which are physical symbols on a paper. Aggregations and relationships of mental objects correspond to sets and to relations, which, in most cases, are parameterized by "indices", corresponding to names, addresses, locations, time instants, etc.. In the sequel we describe some important operations on relations, yielding further relations deduced from the explicitly given ones. Particular and important cases of relations are

functional relations, enabling the representation of conclusions (from x follows $y = f(x)$); partial ordering relations, enabling a grading (valuation) of objects. Applications are multi-valued logics, comparisons of objects a, b with respect to a valued property (p, v) . For examples: event of occurrence of the object at time instant v , $v(a) \leq v(b)$, a happens earlier than b ; occurrence of the object in a fixed sample, measured by the relative frequency v , $v(a) \leq v(b)$, object a occurs less often than object b ; similarity / distance of the object to a fixed object c , measured by an abstract measure v , $v(a) \leq v(b)$, a is less similar / closer to object c than object b . The latter example shows the relationship to topology (in engineering terms: "fuzzy sets").

As commonly applied in mathematics, the representation of knowledge can be facilitated by use of variables.

A symbolisation of a part of interest of our mental knowledge and knowledge processing is named "knowledge base". Given such a knowledge base, "query" operations on it can result in the selection of parts having certain properties, and subject to rules, in the construction of new relations from selected parts, in the test of a given object for being a component or an approximation to a component of the knowledge base, and finding a nearest knowledge base component to the test object.

Like for classical data base systems, query and system definition and manipulation languages could be defined, which is not in the goal of this paper.

2 Relational Structures and Operations

2.1 Hierarchical Relations

For any non-empty sets I, S and any function $\iota: I \rightarrow S$ we use the following notations: ι is an indexing, I is an index set, S is an object set, $s_{[i]} =_{\text{def}} \iota(i)$ (which is an element of S), $s_i =_{\text{def}} (i, s_{[i]})$ (which is an element of $I \times S$). $(s_i)_{i \in I}$ denotes a "family", $\iota(I) = \{s_{[i]} \mid i \in I\} \subseteq S$ is the set of the family $(s_i)_{i \in I}$, parameterized by I .

For $s \in \iota(I)$, the reciprocal image of s , $\iota^{-1}(s)$, is an equivalence class. $(s_i)_{i \in \emptyset}$ denotes the "empty family" ε .

Let there be given a family $(S_i)_{i \in I}$ of non-empty sets $S_{[i]}$, named also "types".

Then the "product set" of this family is defined by $\prod_{i \in I} S_i =_{\text{def}} \{(s_i)_{i \in I} \mid \bigwedge i \in I (s_{[i]} \in S_{[i]})\}$. If all $S_{[i]}$ are identical with S , we write S^I (exponentiation of S by I). The set product is unordered. However, for linearly ordered finite I , e.g. $I = \{1, 2, \dots, n\}$,

the product is isomorphic to the Cartesian product $\times_{i=1,2,\dots,n} S_i$. Classically, any $R \subseteq \prod_{i \in I} S_i$ is a "relation". More general, we define $R \subseteq \bigcup_{U \in \text{pow } I} \prod_{i \in U} S_i$ as relation. An

element of R is then of the form $(s_i)_{i \in U}$. R is a non parameterized set. It can be parameterized by an indexing $\iota: J \rightarrow R$. The result is a family of families, one hierarchical level higher than that of the component families, denoted by $((s_{ji})_{i \in I(j)})_{j \in J}$

with well determined $I[j] \subseteq I$. We assume $\bigcup_{j \in J} I[j] = I$. We define $\bigwedge_{i \in I} (J[i] =_{\text{def}} \{j \mid j \in J \wedge i \in I[j]\})$. Then $R^c = ((s_{ji})_{j \in J[i]})_{i \in I}$ defines the conjugate relation to R . Denoting $I_j =_{\text{def}} \{j\} \times I[j]$ and $J_i^c =_{\text{def}} J[i] \times \{i\}$, we have $M =_{\text{def}} \bigcup_{j \in J} I_j = \bigcup_{i \in I} J_i^c \subseteq J \times I$ with $J = \bigcup_{i \in I} J[i]$. Both, R and R^c can be concatenated to $(s_{ji})_{ji \in M}$. Similarly,

families of families of any hierarchical level and with multiple indices can be defined:

For $v = 1, 2, \dots$ let there be given an index set $I(v) \neq \emptyset$, a set $\{R_{[i]}^{(v-1)} \mid i \in I(v-1)\}$ of non parameterized, non-empty relations of level $v-1$. $R_{[i]}^{(0)}$ is a set of "primitives" (families of level 0). We choose $R^{(v)}$ on level v over the primitives as

$$\emptyset \neq R^{(v)} \subseteq \prod_{i \in I(n)} R_i^{(n-1)}, \text{ and define the general relation}$$

$$\mathcal{R}[n] =_{\text{def}} \bigcup_{v=1,2,\dots,n} R^{(v)}, \text{ containing families of levels up to } n.$$

This construction is in analogy to the one for the sets of sets hierarchy.

2.2 Functions on Relations

Let there be given a universe $\mathcal{U}$ of general, non parameterized relations. For any set $R \in \text{pow } \mathcal{U}$, $R \neq \emptyset$, a function $f: R \mapsto S \in \mathcal{U}$ which lowers the set level is named a "concatenation" of R , denoted by $\mathbf{K}$ (prefix-) or κ (infix-notation). An illustration is $\{\{\dots\}, \{\dots\}, \dots\} \mapsto \{\dots\}$ by set union or intersection. In particular, if $R = \{U\}$, $U \in \mathcal{U}$, i.e. R is a singleton, then $\delta: R \mapsto U$ is a concatenation.

For a universe $\mathcal{F}$ of parameterized relations, i.e. families, concatenations can be defined in analogy. An illustration is: $((a, b, c), (d, (e, f))) \mapsto (a, b, c, d, (e, f))$. Applied to the above relations, we define:

$$\tilde{R}^{(n)} =_{\text{def}} R^{(n)},$$

for $v = n-1, \dots, 1$ and for concatenations $\mathbf{K}$:

$$\tilde{R}^{(v)} =_{\text{def}} R^{(v)} \cup \mathbf{K} \tilde{R}^{(v+1)} \text{ is on level } v.$$

We consider some examples of functions on relations:

(1) For a given $s =_{\text{def}} (s_i)_{i \in I}$ and an index set J , the "projection" of s on J is defined by $pr[J](s) =_{\text{def}} (s_i)_{i \in I \cap J}$, the "co-projection" with respect to J is defined by $cpr[J](s) =_{\text{def}} (s_i)_{i \in I \setminus J}$. For $pr[\{j\}]$ we write also $pr[j]$. We have $pr[J](s) \subseteq s$, $cpr[J](s) = \mathbf{C}[s](pr[J](s))$ ($\mathbf{C}[s]$ means the complement with respect to s). The projection (co-projection) of a set of families is defined by the set of the projections (co-projections) of all families of the set.

(2) $R = ((s_{ji})_{i \in I[j]})_{j \in J} \mapsto F = (s_{ji})_{ji \in M}$ with $M =_{\text{def}} \bigcup_{j \in J} I_j$ (i.e. $M \subseteq J \times I$) is a

concatenation. In turn, the projection of F on $V = \{j\} \times I[j] \subseteq M$ yields $F[V] = (s_{ji})_{i \in I[j]}$.

(3) If $\bigwedge j, j' \in J (j \neq j' \Rightarrow I[j] \cap I[j'] = \emptyset)$ and if we set $I = \bigcup_{j \in J} I[j]$, $s_{[j]i} = s_i$ for $i \in I[j]$, then $((s_{ji})_{i \in I[j]})_{j \in J} \mapsto (s_i)_{i \in I}$ is a concatenation.

(4) We assume: $R' \subseteq \bigcup_{U \subseteq I} \prod_{i \in U} S_i$, $R = ((s_{[j]i})_{i \in I[j]})_{j \in J}$ is a parameterization of R' , J is finite, for each $S_{[i]}$ and each $U, \emptyset \neq U \subseteq J$, a (commutative) function $\#_i: S_{[i]}^U \rightarrow S_{[i]}$ is given with $\#_i((s)) = (s)$ for U being a singleton. Then R can be concatenated to $\mathbf{KR} = (\#_i((s_{[j]i})_{j \in J[i]}))_{i \in I}$. If the set product is Cartesian, $\#_i$ need not be commutative. For example, we take a non-empty set D and $S =_{\text{def}} \text{pow } D$ (the power set of D) with the set operations $\cap, \cup$. We consider the relation $R = ((s_{ji})_{i \in I[j]})_{j=1, 2, \dots, n}$ with all $s_{[j]i} \in S$. The conjugate is $((s_{ji})_{j \in J[i]})_{i \in I}$ with $J[i] \subseteq \{1, 2, \dots, n\}$, $I = \bigcup_{j=1, 2, \dots, n} I[j]$. R can

be concatenated to $(\#_i((s_{[j]i})_{j \in J[i]}))_{i \in I}$, where $\#_i$ represents $\cap$ or $\cup$.

(5) Projections correspond to a selection of a subrelation of a given relation "by address". We can also consider selections "by content" or by "being part of" a given relation. Let there be given $s = (s_i)_{i \in I}$, $s_{[i]} \in S_{[i]}$, $\emptyset \subset U \subset \bigcup_{i \in I} S_{[i]}$. We define the cut

of s along U by $\text{cut}[U](s) = (s_i)_{i \in I[U]}$ with $I[U] = \{i \mid i \in I \wedge \forall u \in U (s_{[i]} = u)\}$. Moreover, let there be given $R = (s_j = (s_{ji})_{i \in I[j]})_{j \in J}$ with a *bijective* parameterization by J , and $t = (t_i)_{i \in I[t]}$ (a "test family") with $s_{[j]i} \in S_{[i]}$, $t_{[i]} \in S_{[i]}$, $\emptyset \subset I[t] \subseteq I$. We define $\text{cut}[t](R) = ((s_{ji})_{i \in I[j]})_{j \in J[t]}$ with $J[t] = \{j \mid j \in J \wedge \bigwedge i \in I[t] (s_{[j]i} = t_{[i]})\}$. t selects $R[t] =_{\text{def}} (s_j)_{j \in J[t]}$. If $J[t] = \{j\}$ then $s_j = (s_{ji})_{i \in I[j]}$ is "characterized" by t .

2.3 Functional Decompositions

We consider the bijectively parameterized relation $R = ((s_{ji})_{i \in I[j]})_{j \in J}$ and a partitioning of all $I[j]$ into two parts, i.e.: $I[j] = I[j]' \cup I[j]''$ with $I[j]' \cap I[j]'' = \emptyset$ and with $I[j]' \neq \emptyset$. We set $x_j =_{\text{def}} (s_{ji})_{i \in I[j']}$, $y_j =_{\text{def}} (s_{ji})_{i \in I[j]'}$, $X =_{\text{def}} \{x_{[j]} \mid j \in J\}$, $Y =_{\text{def}} \{y_{[j]} \mid j \in J\}$ and assume $\bigwedge j, j' \in J (y_{[j]} \neq y_{[j']} \Rightarrow x_{[j]} \neq x_{[j']})$. In this case we say the partition is functional and yields a function $f: X \xrightarrow{\text{onto}} Y$ with $x_{[j]} \mapsto_R y_{[j]}$ according R . In the following, $\mapsto_R$ is replaced by $\mapsto$ if the meaning is clear from context. We

use the same notation f for the functional relation of all pairs $(x_{[j]}, y_{[j]}), f \subseteq X \times Y$.

From $x_{[j]} \mapsto y_{[j]}$, $u_{[j]}$ being a projection of $y_{[j]}$ and $v_{[j]}$ being its co-projection with respect to $y_{[j]}$, i.e. $y_{[j]} = u_{[j]} \ltimes v_{[j]}$, follow the functional assignments $x_{[j]} \mapsto u_{[j]}$, $x_{[j]} \mapsto v_{[j]}$ and $x_{[j]} \ltimes u_{[j]} \mapsto v_{[j]}$. A functional partition $\{(x_{[j]} \mapsto y_{[j]}) \mid j \in J\}$ is named "coarser" than a functional partition

$\{(x'_{[j]} \mapsto_g y'_{[j]}) \mid j \in J\}$, if for all $j \in J$ $x_{[j]} \subseteq x'_{[j]}$. Thus $\{(x_{[j]} \mapsto_f y_{[j]}) \mid j \in J\}$ is coarser than $\{(x_{[j]} \mapsto_g u_{[j]}) \mid j \in J\}$. Coarseness defines a partial ordering of the functional partitions of R .

Given $x_{[j]} \mapsto y_{[j]}$ with respect to f , then the reciprocal image of $y_{[j]}$ is $f^{-1}(y_{[j]}) =_{\text{def}} \{x_{[j]} \mid j \in J \wedge f(x_{[j]}) = y_{[j]}\}$. $f^{-1}(y_{[j]})$ defines an equivalence class on $X(f)$, the domain of f . For f coarser g we say also g "finer" f . We have relationships $x_{[j]} \mapsto y_{[j]} \mapsto u_{[j]} \mapsto_g (u_{[j]})$. Application: If $x_{[j]i}$ is interpreted as a "property" of "object" $y_{[j]}$, $x_{[j]} = (x_{[j]i})_{i \in I[j]}$ as the "composite property" of the object $y_{[j]}$ to possess all properties $x_{[j]i}$, then the object is uniquely determined by this composite property.

Let $X(f) = (x_j)_{j \in J}$ be bijectively parameterized by J and let $t = (t_i)_{i \in I[t]}$ with $I[t] \subseteq \bigcup_{j \in J} I'[j]$ be a test family. $\text{cut}[t]X(f)$ selects $(x_j)_{j \in J[t]}$ and according f the family

$(y_j = f(x_{[j]j}))_{j \in J[t]}$. For a test family $t' \subseteq t$ we have $(y_j)_{j \in J[t]} \supseteq (y_j)_{j \in J[t']}$. Application: The less common properties, the larger is the set of objects possessing these properties.

Functional relationships implicitly given by R are for example:

$t \mapsto (y_j \text{ for } j \in J[t]) \mapsto (x_{j*} \text{ for } x_{[j]*} \in f^{-1}(y_{[j]}))$ and for $j \in J[t] \mapsto \mathbf{C}[x_{j*}](t)$ for all x_{j*} , and the same recursively for the test family $t' =_{\text{def}} \mathbf{C}[x_{j*}](t)$.

2.4 Object- and Indextransformations

Let there be given: $R[U] \subseteq \prod_{i \in U} S_i$, a finite index set J for bijective indexing, and

for all $j \in J$, $s_{[j]} \in R[U]$, $s_j = (s_{ji})_{i \in U}$. If for all i a composition law $\oplus_i: S_i^J \rightarrow S_i$ is given, we can define $\oplus_J((s_j)_{j \in J}) =_{\text{def}} (\oplus_i((s_{ji})_{j \in J}))_{i \in U}$, which is a concatenation.

We consider $s = (s_i)_{i \in U} \in \prod_{i \in U} S_i$, $t = (t_k)_{k \in V} \in \prod_{k \in V} T_k$, and we assume $V = \tau(U)$

for a given mapping $\tau: U \rightarrow V$. This defines the functional relationship $\wedge k \in V (\{s_i \mid i \in \tau^{-1}(k)\} \mapsto \{t_k\})$. Particular cases are: $t_k = \mu(\{s_i \mid i \in \tau^{-1}(k)\})$; τ being 1:1, $S_{[i]} = T_{[\tau(i)]}$, $s_{[i]} = t_{[\tau(i)]}$. Applications are: μ being a measure on $\bigcup_{i \in U} S_i$; τ being a

movement of a geometrical object in Euclidean space.

3 Valuation of Relations

In many applications, the families $s^{(n+1)} = (s_i^{(n)})_{i \in I}$ of a hierarchical relation are "valuated", i.e. on level n a value $v_{[i]}^{(n)}$ taken of a domain $V^{(n)}$ is assigned to $s_i^{(n)}$. This functional assignment is represented by $(s_i^{(n)}, v_i^{(n)})_{i \in I}$ and by $(s^{(n+1)}, v^{(n+1)})$. The

valuations may be level dependent, but are in general independent of each other and usually assumed not to depend on the objects valued. A particular case is the existence of a functional dependence $v^{(n+1)} = \varphi((v_i^{(n)})_{i \in I})$.

We explain this in detail: For levels $n = 0, 1, 2, \dots$ let there be given an index set $I^{(n+1)} \neq \emptyset$, for all $i \in I^{(n+1)}$ a set of relations $S_{[ij]}^{(n)} \neq \emptyset$, and a set of valuations $V_{[ij]}^{(n)}$. Any chosen $S^{(n+1)}$, $\emptyset \neq S^{(n+1)} \subseteq \prod_{i \in I^{(n+1)}} (S_i^{(n)} \times V_i^{(n)})$ is a relation on level $n+1$. We

consider a selected $S^{(n+1)}$ and assume, this relation is bijectively parameterized by the elements of a set J . Any pair (j, i) determines (s_{ji}) and v_{ji} uniquely, thus for given parameterizations, $(s_{ji}, v_{ji})_{i \in I[j]}$ can be represented by $(v_{ji})_{i \in I[j]}$.

Reducing the generality, we assume for all n and i that on level n all valuation types $V_{[ij]}^{(n)}$ are equal to a type $V^{(n)}$. In many applications, the $V^{(n)}$ are lattices or complete lattices $(V^{(n)}, \leq_{(n)})$ with additional structures. This case is treated in the following:

For a complete lattice $(V, \leq, \sqcup, \sqcap)$ and an index set $\mathcal{I}$, we consider $\tilde{V} \stackrel{\text{def}}{=} \bigcup_{\emptyset \subset I \subset \mathcal{I}} V^I$ with elements $v = (v_i)_{i \in I}$, $v_{[ij]} \in V$. We want a reflexive and transitive extension " $\prec$ " of " $\leq$ " as given on V to $\tilde{V}$, i.e. $\prec = \leq$ on V . We define for $v = (v_i)_{i \in I} \in \tilde{V}$, $v' = (v'_j)_{j \in J} \in \tilde{V}$: $v \prec v' \Leftrightarrow \bigwedge i \in I (\bigvee j \in J (v_{[ij]} \leq v'_{[jj]}))$. $\prec$ is in general no order relation, from $v \prec v'$ and $v' \prec v$ need not follow $v = v'$. Given another complete lattice $(W, \leq, \sqcup, \sqcap)$, $W \cap V \neq \emptyset$ is admitted, then $\prec$ -homomorphisms $\lambda: V^* \subseteq \tilde{V} \rightarrow W$ (named "logic functions" in [1]) are considered. In this case we have $(v \prec v') \Rightarrow (\lambda v \leq \lambda v')$, $(\sqcup v \leq \sqcup v') \Rightarrow (\lambda \sqcup v \leq \lambda \sqcup v')$, $(v \prec v' \text{ and } v' \prec v) \Rightarrow (\lambda v = \lambda v')$, $\sqcap v \prec v \prec v' \prec \sqcup v' \Rightarrow \lambda \sqcap v \leq \lambda v \leq \lambda v' \leq \lambda \sqcup v'$ and for $v = v'$: $\lambda \sqcap v \leq \lambda v \leq \lambda \sqcup v$. In case $v = (v_i)_{i \in I}$ and all $v_{[ij]}$ are equal to the element e , then $\lambda e = \lambda v$. So far, for $w = \lambda((v_i)_{i \in I})$, λ depends on I . If only finite sets I are admitted, λ may depend on $\text{card } I$ only, i.e. $\lambda((v_i)_{i \in I}) = \lambda((v_{\beta(i)})_{i \in I})$ for any bijection $\beta: I \rightarrow J$ with $v_{[ij]} = v_{[\beta(i)]}$.

Examples for relations $u = (u_i)_{i \in I} \prec v = (v_i)_{i \in J}$ and associated λ -functions are:

$u \prec v$	defined by	$\lambda(\prec)$
$\prec_{\wedge v}$	$\bigwedge u_i \in u \bigvee v_j \in v (u_{[ij]} \leq v_{[jj]})$	$\sqcup$
$\prec_{v \wedge}$	$\bigwedge v_j \in v \bigvee u_i \in u (u_{[ij]} \leq v_{[jj]})$	$\sqcap$
$\prec_{\wedge v \wedge}$	$\prec_{\wedge v} \wedge \prec_{v \wedge}$	$\sqcup, \sqcap$
$\prec_{\wedge \wedge}$	$\bigwedge u_i \in u \bigwedge v_j \in v (u_{[ij]} \leq v_{[jj]})$	$\sqcup, \sqcap$
$\prec_\beta$	it exists $\beta: I \xrightarrow{1:1} J$ such that $\bigwedge u_i \in u (u_{[ij]} \leq v_{[\beta(i)]})$	$\sqcup, \sqcap$

A particular case of $\prec_\beta$ is the component wise partial ordering of vectors in an n -dimensional vector space over the field $\mathbf{R}$.

Analogously, $\prec$ - antimorphisms can be defined as λ - functions.

Important examples are:

(1) $V = W = (\mathbf{R}_0$ (the non-negative real numbers), $\leq, +, *$), $\mathfrak{I} = \mathbf{N}_0$ (the natural numbers with 0), finite $I \subset \mathfrak{I}$. $v = (v_i)_{i \in I} \in V^I$, $\omega = (\omega_i)_{i \in I} \in V^I$, used as "weights", $n = 1, 2, 3, \dots$:

$$\lambda[\Sigma, n] \omega v =_{\text{def}} \sum_I \omega_{[i]} v_{[i]}^n,$$

$$\lambda[\Sigma, n, \text{mean}] \omega v =_{\text{def}} \left(\sum_I \omega_{[i]} v_{[i]}^n \right) / \sum_I \omega_{[i]},$$

$$\lambda[\Sigma, \omega, n, \text{mean, normed}] \omega v =_{\text{def}} \left(\sum_I \omega_{[i]} v_{[i]}^n \right)^{1/n} / \left(\sum_I \omega_{[i]} \right)^{1/n},$$

$$\lambda[\max] v =_{\text{def}} \max_I \{v_{[i]} \mid i \in I\}, \lambda[\min] v =_{\text{def}} \min_I \{v_{[i]} \mid i \in I\},$$

$$\lambda[\alpha \max + (1 - \alpha) \min] v =_{\text{def}} \alpha \lambda[\max] v + (1 - \alpha) \lambda[\min] v \text{ for } 0 \leq \alpha \leq 1.$$

(2) $V = W = (\mathbf{R}_1$ (the real numbers ≥ 1), $\leq, +, *$), else like above.

$$\lambda[\Pi] \omega v =_{\text{def}} \prod_I v_{[i]}^{\omega_{[i]}},$$

$$\lambda[\Pi, \text{normed}] v^\omega =_{\text{def}} \left(\prod_I v_{[i]}^{\omega_{[i]}} \right)^{\text{exp}} \text{ with } \text{exp} = 1 / \sum_I \omega_{[i]}$$

Let be $v = (v_i)_{i \in I}$, $u = (u_j)_{j \in J}$, $\alpha = (\alpha_i)_{i \in I}$, $\beta = (\beta_j)_{j \in J}$, $\kappa(+)$ and $\kappa(*)$ concatenations for case (1) and case (2), respectively. Examples for "triangle inequalities" are:

$$\lambda[\Sigma, n, \text{mean}] (\alpha v \kappa(+)) \beta u \leq \lambda[\Sigma, n, \text{mean}] (\alpha v) + \lambda[\Sigma, n, \text{mean}] (\beta u),$$

$$\lambda[\Pi, \text{normed}] (v^\alpha \kappa(*) u^\beta) \leq \lambda[\Pi, \text{normed}] (v^\alpha) * \lambda[\Pi, \text{normed}] (u^\beta).$$

Example (propositional calculus): Let be $\mathfrak{I} = \mathbf{N}_0$, for all levels, $V^{(n)} = (\{f, t\}, f < t)$ (type Boolean), $S^{(0)} = \{a, b, c, \dots\}$ a set of propositions. λ -functions are $\wedge, \vee$ (homomorphisms), $\neg \wedge, \neg \vee$ (antimorphisms). For example

level 0: $\{a, b, c, \dots\}$,

level 1: $\{(a, t), (b, f), (c, t)\}$, $\text{card } I^{(1)} = 3$,

level 2: $\{(((a, t), (b, f)), \wedge(t, f) = f), (((a, t), (c, t)), \vee(t, t) = t)\}$, $\text{card } I^{(2)} = 2$.

Indices are suppressed.

Example (multi-valued propositions): Set of propositions: $P = \{(p_1 = \text{"sun is shining"}), (p_2 = \text{"it rains"})\}$. Set of values: $V(3) = \{0, .5, 1\}$, 3-valued logic. The relation of binary compositions of propositions is presented by the binary relation of their valuations, $((p_2, v_j), (p_1, v_i)) \leftrightarrow (v_j, v_i)$, see Fig. 1.

Because of semantic reasons, the set $\overline{\mathcal{P}}$ of propositions with entries $\{(3, 3) \leftrightarrow (\text{full sun, full rain}), (3, 2) \leftrightarrow (\text{full sun, partly rain}), (2, 3) \leftrightarrow (\text{partly sun, full rain})\}$ is declared to be impossible, the complementary set $\mathcal{P}$ is possible, $\mathcal{R} = \mathcal{P} \cup \overline{\mathcal{P}}$, $\mathcal{P}$ can be valued by "1", $\overline{\mathcal{P}}$ by "0", λ is only defined on $\mathcal{P}$. An interpretation of the λ -values could be: 0 unpleasant, .5 fair, 1 fine weather.

j	V_2				
3	1	(1, 0), 0 $\lambda = \min$	(1, .5)	(1, 1)	
2	.5	(.5, 0), 0 $\lambda = \min$	(.5, .5), .5 $\lambda = \text{any}$	(.5, 1)	
1	0	(0, 0), 0 $\lambda = \text{any}$	(0, .5), .5 $\lambda = \max$	(0, 1), 1 $\lambda = \max$	
		0	.5	1	V_1
		1	2	3	i

Fig. 1. Relation $\mathcal{R}$ of logic values, entries (v_i, v_j) , $\lambda(v_i, v_j)$

4 Topological Structures on Relations

To express neighborhood / similarity relations on the families of a relation $R \subseteq \bigcup_{U \in \text{pow } I} \prod_{i \in U} S_i$, we introduce topological structures by means of "filter bases": Given a

set S , then $(\text{pow } S, \subseteq, \cup, \cap)$ is a complete Boolean lattice. Let $B_{[0]}, \emptyset \neq B_{[0]} \subseteq S$, be a set of points which by assumption are all equivalent with respect to the admitted accuracy. Further, let $\mathfrak{B} = \{B_{[k]} \mid B_{[k]} \subseteq S \text{ and } k \in K\}$ be a set of sets with the properties: from $B_{[k']} \in \mathfrak{B}$ and $B_{[k'']} \in \mathfrak{B}$ follows, it exists a $B_{[k]} \in \mathfrak{B}$ with $B_{[k]} \subseteq B_{[k']}$ and $B_{[k]} \subseteq B_{[k'']}$. Then $\mathfrak{B}$ is a "filter base". We assume: $B_{[0]} = \bigcap \mathfrak{B}$ (denoted $\lim \mathfrak{B}$), $S \supseteq B =_{\text{def}} \bigcup \mathfrak{B}$ (support of $\mathfrak{B}$). We further assume, $\mathfrak{B}$ is a complete lattice.

Then for $s \in S$, $s^* \in \lim \mathfrak{B}$, "generalized distances" $d_{\cap}(s^*, s) =_{\text{def}} \bigcap \{B \mid B \in \mathfrak{B} \text{ and } s \in B\} \in \mathfrak{B}$ and $d_{\cup}(s^*, s) =_{\text{def}} \bigcup \{B \mid B \in \mathfrak{B} \text{ and } s \notin B\} \in \mathfrak{B}$ can be defined. Let there be given a complete lattice $(V, \leq, \sqcup, \sqcap)$ and a $\leq$ -homomorphism $\phi: \mathfrak{B} \rightarrow V$, defining a valuation of $\mathfrak{B}$. We use $v(s^*, s) =_{\text{def}} \phi(d_{\cap}(s^*, s))$ as a distance measure of s from s^* . "Fuzzy sets" in the engineering literature are examples of filter bases, generated by the reciprocal image $\overset{-I}{\phi}$ of $\overset{\text{onto}}{\phi}: S \rightarrow \{[\alpha, 1], 0 \leq \alpha \leq 1\}$.

Let $\{\mathfrak{B}_{[l]} = \{B_{[lk]} \mid k \in K\} \mid l \in L\}$ be a set of isomorphic filter bases, and for all k , let $B_{[lk]} \in \mathfrak{B}_{[l]}$ with respect to l be uniformly valued by a $\subseteq$ -homomorphism ϕ into V , i.e. $\phi(B_{[lk]}) = \phi(B_{[l'k]})$. We further assume, for $l \neq l'$ holds $\lim \mathfrak{B}_{[l]} \cap \lim \mathfrak{B}_{[l']} = \emptyset$. Then the $\mathfrak{B}_{[l]}$ are uniform neighborhood systems to the separated sets $\lim \mathfrak{B}_{[l]}$. For more details we refer to [1, 2, 3, 4, 5].

We apply these topological concepts to the relation R : Let $R^* \subseteq R$ be a non-empty subset. We set $S = R$, $L = R^*$. To each $r^* \in R^*$ let be adjoined a

neighborhood system in form of a filter base $\mathfrak{B}_{[r^*]}$ on $\text{pow } R$ as defined above, with $r^* \in \lim \mathfrak{B}_{[r^*]}$, and with all sets $\lim \mathfrak{B}_{[r^*]}$ being disjoint. A trivial case is, $R = R^*$, $\lim \mathfrak{B}_{[r^*]} = \{r^*\}$.

For fixed r^* , $v(r^*, r) =_{\text{def}} \phi(d_{\cap}(r^*, r))$ defines an abstract distance of any $r \in B_{[r^*]} =_{\text{def}} \bigcup \mathfrak{B}_{[r^*]}$ from any $r^* \in \lim \mathfrak{B}_{[r^*]}$. Hence, to any $r^* \in \lim \mathfrak{B}_{[r^*]}$ exists a neighborhood $N(r^*)$ with elements $u \in R$ "nearer" to r^* than to any other comparable $\hat{r} \in R^* \setminus \lim \mathfrak{B}_{[r^*]}$. If the relation R is hierarchical, then the topological reasoning can be applied to each hierarchical level.

We revisit section 2.2 and apply the preceding to $R^* \subseteq X(f)$, i.e. to each $x_{[j]} \in R^*$ exists by assumption a neighborhood system $\mathfrak{B}_{[j]}$. Let there be given a test family $t \in X(f)$. For $t \in B_{[j]}$, $d_{\cap}(x_{[j]}, t)$ and $v(x_{[j]}, t) = \phi(d_{\cap}(x_{[j]}, t))$ exist. We evaluate $y = f(t)$ by $v(x_{[j]}, t)$. If $y \in N(f(x_{[j]}))$, we classify y to $y_{[j]} = f(x_{[j]})$ with valuation $v(x_{[j]}, t)$. Applications are pattern classifiers and artificial neural networks.

References

- 1 Albrecht R. F.: On mathematical systems theory. In: R. Albrecht (ed.): Systems: Theory and Practice, Springer, Vienna-New York, (1998) 33–86
- 2 Albrecht R. F.: Topological Approach to Fuzzy Sets and Fuzzy Logic. In: A. Dobnikar, N. C. Steele, D. W. Pearson, R. F. Albrecht (eds.): Artificial Neural Nets and Genetic Algorithms, Springer, Vienna-New York (1999) 1–7
- 3 Albrecht R. F.: Topological Theory of Fuzziness. In: B. Reusch (ed.): Computational Intelligence, Theory and Applications, LNCS vol. 1625, Springer, Heidelberg-New York (1999) 1–11
- 4 Albrecht R. F.: Topological Concepts for Hierarchies of Variables, Types and Controls. In: G. Alefeld, J. Rohn, S. Rump, F. Yamamoto (eds.): Symbolic Algebraic Methods and Verification Methods, Springer Vienna-New York, (2001) 3–10
- 5 Albrecht R. F., Németh G.: A Generic Model for Knowledge Bases, Periodica Polytechnica, Electrical Engineering, TU Budapest, vol. 42 (1998) 147–154

How Many Rounds to KO?, or Complexity Increase by Cryptographic Map Iteration

Juan David González Cobas and José Antonio López Brugos

Computer Science Department
Universidad de Oviedo
`{cobas,brugos}@etsiig.uniovi.es`

Abstract. Iterating a highly non-linear mapping is the basis of the classic schema for building block ciphers, in the form of *Feistel networks*. The number of rounds of such constructions is a critical parameter. In this paper, the number of rounds needed to reach a certain minimum complexity bound is proposed as a valid measure to assess the cryptographic significance of certain boolean functions. The most remarkable facts arising from this approach are the dependency of the number of rounds on some predefined weaknesses of the tested functions, and the failure to pass the proposed tests when complexity measures are chosen *ad hoc* to address those weaknesses.

1 Feistel Networks and Iterated Maps

A succesful schema for the design of block ciphers is the *Feistel network*. This kind of construction operates by iterating a transformation over the state of a register in a cleverly chosen way, allowing for easy inversion and introduction of key dependency.

A graphical description of this device is given in Fig. 1. The basic operation divides the plaintext information in two equal-sized halves L and R , and applies a nonlinear function f to them according to the depicted schema, obtaining new values for the registers L and R . Algebraically

$$L' = R \tag{1}$$

$$R' = L \oplus f(K_i, R) \tag{2}$$

The key-dependent information K_i is extracted from the whole key of the cipher algorithm according to the *key schedule algorithm*.

A complete design for a block cipher specifies

1. A key schedule algorithm $S(K, i) \rightarrow K_i$ extracting some bits K_i from the key K in a round-dependent fashion.
2. A function f , cleverly designed to interact well with itself and the key schedule to achieve optimal confusion and diffusion characteristics [1].
3. A number of rounds, i.e., how many times the schema of Fig. 1 is iterated.

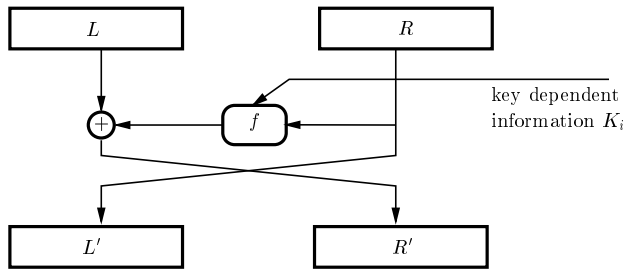


Fig. 1. Operation of a Feistel network

As a classical example, the most successful and popular design of a Feistel-type block cipher, the Data Encryption Standard (DES), specified sixteen rounds of a network whose nonlinear part f embodied (as per 1979 standards) rather intriguing design criteria [2].

It turns out that there is a strong interaction among the design of the substitution-permutation network of DES, the key-schedule algorithm and its very numbers of rounds. The magic figure of sixteen is indeed the lowest one to make DES resilient to differential cryptanalysis, a fact ignored in the open until 1991. However, this was taken into account in the design, according to [3]

2 How Many Rounds to KO?

As it turns out, the number of rounds in a block cipher is a critical parameter, once the structure of the substitution-permutation network is chosen. A theoretical interpretation of this dependency can be given in the context of the Markov cipher theory [4]. A rough description of this interpretation can be made in complexity-theoretic terms.

When a plaintext block is submitted to a block cipher whose structure matches the principles stated in Section 1, the output of every round displays (ideally) a complexity profile which converges rapidly to the expected maximum complexity per symbol (see Fig. 3 for some typical curves, specially the one labelled “Serpent”). Even when the initial point of the process is highly redundant (i.e. has a low complexity profile), repeated application of the round function increases entropy (or whatever measure of complexity suits the application) until the statistically expected entropy of a random source is reached.

This suggests a complexity-theoretical means of testing the performance of a proposed round function $F(K, x)$ as far as its diffusion and confusion characteristics are concerned. For this doing, a very low complexity stream of (plaintext) data $\{M_n\}^{(1)}$ is initially considered. A canonical example is $M_n^{(1)}$ identically 0

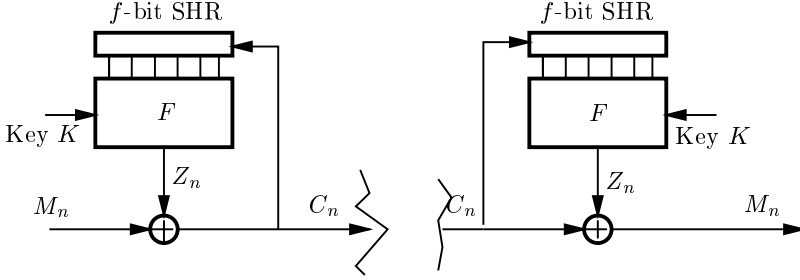


Fig. 2. Basic self-synchronizing stream cipher (CTAK mode)

for a suitable block size (depending on the cryptographic function considered), but nonconstant streams could be considered as well. In each iteration, it is transformed according to

$$\begin{aligned} M_n^{(i+1)} &= M_n^{(i)} & \text{for } n < i \\ M_n^{(i+1)} &= F(K, M_n^{(i)}) & \text{for } n \geq i \end{aligned}$$

so that further elements of the stream eventually undergo the action of an increasing number of rounds. Then the complexity profile (linear, maximum order or whatever) of the resulting stream $M_n^{(L)}$ is computed, and normalized by sequence length. The convergence to the limit value (usually 1) has to be as fast as possible. The number of rounds to reach a threshold level $1 - \alpha$, with $\alpha \ll 1$, measures how many iterations it takes the proposed F to “knock out” the redundancy of the original plaintext. A high number of rounds typically reveals a blatant weakness in function design.

3 Some Families of Boolean Key-Dependent Functions

The technique outlined in Sec. 2 has been applied to a special kind of functions arising in the domain of stream cipher design.

A *self-synchronizing stream cipher* (SSSC for short) basic schema is depicted in Fig. 2. In this kind of cryptographic device, the cipher transformation (which usually operates bitwise) maps a plaintext $\{M_n\}$ into a ciphertext $\{C_n\}$ sequence according to the formula

$$C_n = M_n \oplus F(K, C_{n-1}, \dots, C_{n-f}) \quad (3)$$

with proper initial vectors. Deciphering is achieved by inverting equation (3).

The dependency on the f -bit shift register makes recovery from a loss of synchronization automatic after reception of f bits of noise. The design of suitable,

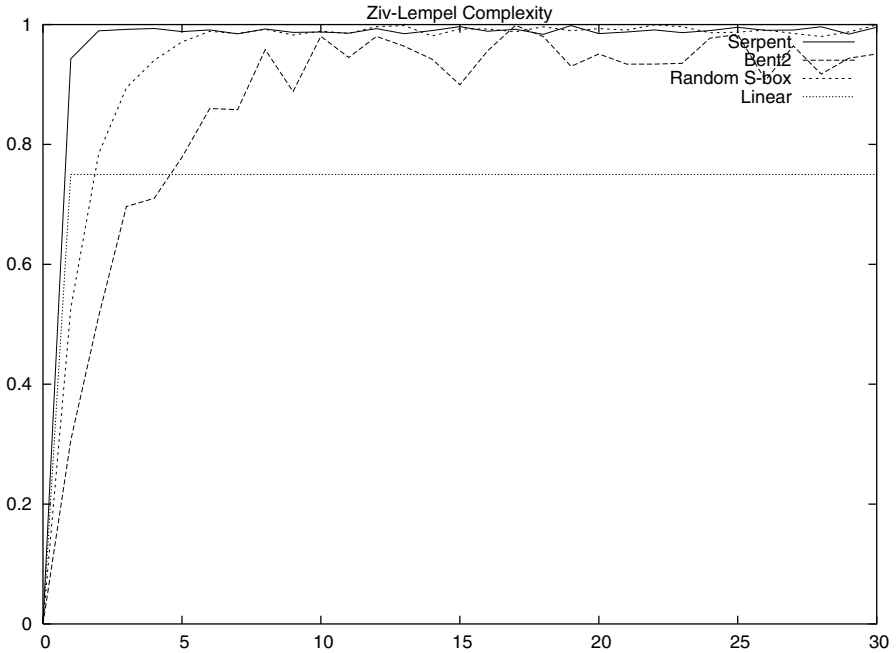


Fig. 3. Complexity profiles for *Bent1*

key-dependent feedback functions for this kind of construction is a difficult and overlooked subject with challenges of its own [5,6].

In [7], some techniques to inject key-dependent information into a SSSC feedback function are proposed and tested. All of them take bent functions as a basis, together with elementary relations for constructing them [8,9]. The procedures depend on recursive application of the following

Theorem 1. *If a, b, c are bent functions of n arguments*

$$f(x, x_{n+1}, x_{n+2}) = a(x) + (c(x) + a(x))x_{n+2} + x_{n+1}x_{n+2}$$

$$g(x, x_{n+1}, x_{n+2}) = b(x) + (b(x) + a(x))(x_{n+1} + x_{n+2}) + x_{n+1}x_{n+2}$$

are bent functions of $n + 2$ arguments

The intended key dependency is introduced by three main devices:

1. At each step of the recursion stated in Theorem 1, the next key bit is used to decide which one of the formulas is to be applied. In other words, the key K constitutes a “program” to the multiway recursion.
2. Precomputing a pseudorandom permutation $\pi_K \in S_n$ by composing transpositions selected according to the bits of K . Then, for a fixed bent, n -ary B , we set $F(K, \{x_i\}) = B(\{x_{\pi_K(i)}\})$. That is, the inputs to B are permuted by π prior to evaluation.

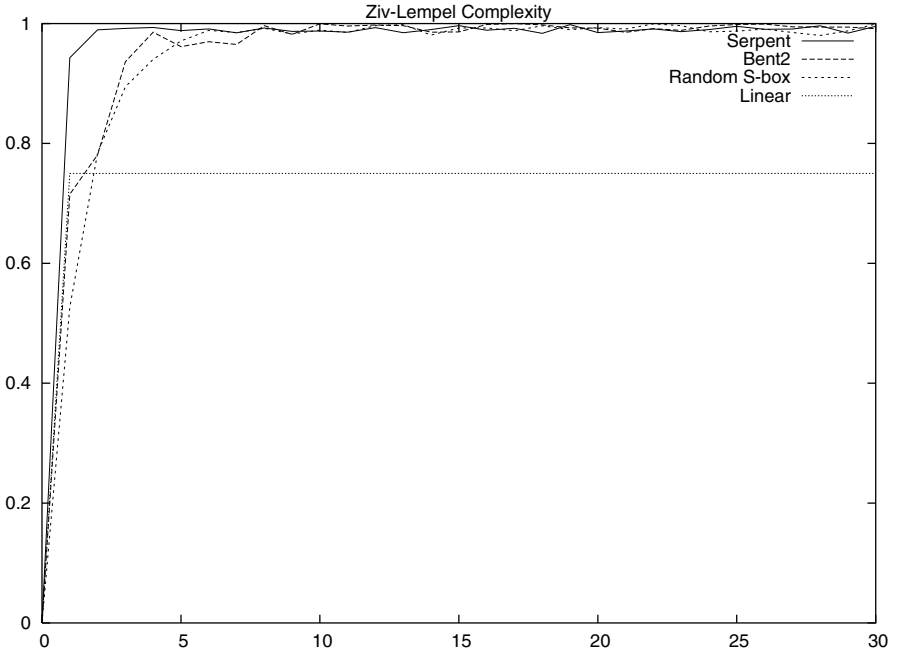


Fig. 4. Complexity profiles for *Bent2*

3. Inserting K linearly into the argument, i.e., $F(K, x) = F(K \oplus x)$.

The analyses in [7] show that the latter is the weakest of the approaches. Observe that in any case, the function obtained is a bent function of its argument (though not necessarily of the key K).

How do the functions obtained by the three approaches behave as far as diffusion and confusion are concerned? Studying these maps as round functions of a hypothetical block cipher, and measuring their required number of rounds to KO seemed natural in the context of self-synchronizing constructions. In the next section we study some of the results obtained.

4 Inferring Weaknesses from KO Tests

In Fig. 3, four complexity profiles are shown. The curve labeled *Bent1* refers to a feedback function defined according to our second method (i.e., using the key as a permutation of the arguments). As a means of comparison, we depict three other test functions:

1. A linear function, whose complexity profile becomes constant after a finite number of rounds.
2. A round function derived from the Serpent block cipher [10]. This is chosen as a reference, given its outstanding power of confusion of a single Serpent round.

3. A round function based on randomly chosen S-boxes, the parameters of which are chosen to be as close as possible to the bent function being tested, according to the guidelines of [11].

The averaged results shown in Fig. 3 are a bit disquieting. There is a good reason for the poor profile exhibited: the complexity measure applied (Ziv-Lempel) shows to be specially discriminating of patterns that this specific design tends to generate. The effect of key dependency amounts only to a constant permutation of ciphertext bits prior to applying the feedback transformation, and this has a deadly impact in the performance shown in this test by the construction proposed.

Much more satisfactory results are given in Fig. 4, where label *Bent2* refers to our construction using the key as a “program” to construct the actual feedback function. Though far from the superb performance of a round function of Serpent, this construction exhibits a slightly better confusion characteristic than a randomly chosen S-box.

It is an interesting fact that past analyses did not reveal substantial differences among the two constructions tested, which these did. Those previous studies were based on standard complexity profile analyses of generated pseudo-random streams, which did not address the specific structure of the constructions involved. In the present setting, the use of Ziv-Lempel was crucial, for better discriminating the weakness of the functions based on a constant permutation of the feedback inputs.

References

1. Shannon, C.E.: Communication theory of secrecy systems. *Bell Sys. Tech. J.* **28** (1949) 657–715
2. National Bureau of Standards: Data Encryption Standard. U. S. Department of Commerce, Washington, DC, USA. (1977)
3. Coppersmith, D.: The Data Encryption Standard (DES) and its strength against attacks. *IBM Journal of Research and Development* **38** (1994) 243–250
4. Lai, X., Massey, J.L., Murphy, S.: Markov ciphers and differential cryptanalysis. *Lecture Notes in Computer Science* **547** (1991) 17–38
5. Rueppel, R.: Analysis and Design of Stream Ciphers. Springer-Verlag, Berlin (1986)
6. Maurer, U.M.: New approaches to the design of self-synchronizing stream ciphers. *Lecture Notes in Computer Science* **547** (1991) 458–471
7. González-Cobas, J.D.: Funciones booleanas con clave para cifrados en flujo autosincronizantes. In: *Actas de la VI Reunión de Criptología y Seguridad de la Información*, Madrid, Ra-Ma (2000) 99–106
8. Rothaus, O.: On bent functions. *Journal of Combinatorial Theory Series A*, **20** (1976) 300–305
9. Meier, W., Staffelbach, O.: Nonlinearity criteria for cryptographic functions. In: *Advances in Cryptology – EUROCRYPT ’89 Proceedings*. *Lecture Notes in Computer Science* 434, Springer-Verlag (1990) 549–562
10. Anderson, R., Biham, E., Knudsen, L.: Serpent: A proposal for the Advanced Encryption Standard. Nist aes proposal, National Institute for Standards and Technology, Gaithersburg, MD, USA (1998)

11. Adams, C., Tavares, S.: Good-S-boxes are easy to find. In Brassard, G., ed.: Proc. CRYPTO 89, Springer-Verlag (1990) 612–615 Lecture Notes in Computer Science No. 435.
12. Schneier, B.: Self-study course in block cipher cryptanalysis. Technical report, Counterpane Systems, 101 East Minnehaha Parkway, Minneapolis, MN 55419 (1999)

A Non-standard Genetic Algorithm Approach to Solve Constrained School Timetabling Problems

Cristina Fernández¹ and Matilde Santos²

¹Lab. Electrónica y Automática. CIEMAT
Avda. Complutense, 22. 28040. Madrid – Spain
`cristina.fernandez@ciemat.es`

²Dto. Arquitectura de Computadores y Automática
Facultad de CC. Físicas – UCM. 28040. Madrid – Spain

Abstract. In this paper a non-standard constructive evolutionary approach is described to solve efficiently timetabling problems, so necessary at every school and university. The basic scheme of Genetic Algorithms has been used, but some operators have been adapted to the characteristics of the problem under consideration, improving its resolution capability. Although the design requires more effort, it helps to reduce the search space. Feasible timetables have been considered as hard constraints in the problem formulation. It has been proved that modified genetic algorithms can be very useful in this kind of problems for the easiness of including new constraints and the effectiveness of the resolution, exhibiting a good compromise between computational time and optimal reaching.

1 Introduction

Timetabling is one of the most common problems at every educational Institution. Every School, College or University throughout the world has to design every year feasible timetables that have to fulfill many requirements. Most of the times this demanding task is carried out manually or with the help of administration systems. In any case, it is a tedious and laborious task. Computational resources appear as a good choice in helping with this repetitive and time consuming task, as an adequate algorithm might guarantee the problem can be solved in reasonable computing time.

On the other hand, from the Artificial Intelligence point of view there has been a large concentration of efforts on university timetabling problems [1, 2], in contrast with school courses and exams scheduling, where there is currently no evolutionary school timetable software available [3], despite some attempts [4, 5]. This is despite the fact that there are several non-evolutionary school timetabling software available, as this problem has been tackled with operation research, linear programming, network flow, etc.

One of the main problem that appears when dealing with scheduling optimization is how to introduce many constraints in the algorithm and how to assign them the appropriate weights. Both hard and soft restrictions must be fulfilled but in a different level. Besides, computational time is usually very high. Evolutionary computing lends itself to multi-constrained problems, due to the facility of incorporating constraint violations into a single fitness function. These intelligent strategies can provide optimal solutions exploring a narrower search space, saving computational time [6].

Course timetabling, as other problems of scheduling, is a multi-dimensional NP-Complete problem [4, 7] as it cannot be solved in polynomial time by the exhaustive evaluation of every timetable. Course timetabling is basically a combinatorial problem [8], with a set of items that have to be sorted to obtain the best configuration. Nevertheless the fitness function in the timetabling problem is quite complicated, needing to take into account many restrictions with different priority.

Genetic Algorithms, as well as other heuristic methods, have been found to be a very powerful optimization tool for NP-complete problems [6, 9, 10]. They explore the solutions space moving towards the most promising areas of the search space. As other heuristics methods, they cannot insure that the final solution is the unique optimal, so they keep different solutions avoiding local minimums or cycling.

In fact, timetabling problems do not usually require an unique optimal solution, but one that satisfies every constrain. Therefore there may be multiple global optimums in the problem and that does not interfere with genetic algorithms solving method. Hard and soft constrains can be easily included in the fitness function that evaluates the kindness of every possible solution. Different weights can be assigned to each of the restriction to guide the searching. There is certain randomness in finding the solution that helps to avoid stagnation in local optimums [11, 12].

The variables involved in a specific timetable configuration are restricted to a discrete and finite set, and the kindness of a specific solution is measured by the order of these variables, which are strongly interdependent. This interrelationship is the main restriction that the solution has to meet, otherwise it is absolutely a non feasible timetable. This important characteristic forces to implement the resolution method so it is adjusted to this constraint. Therefore, GA can be an useful strategy for solving these problems but they are more efficient if they are adapted to the application [13].

In this work, a typical timetabling problem has been implemented. The application considers timetables for one scholar year with different courses, teachers, hours per week, etc. A graphical user interface has been developed to introduce the main characteristics and requirements of the problem, including hard and soft constraints.

Time consumption has been investigated, analysing the number of iterations needed to reach the optimum as the size of the problem increases. It has been proved that the penalisation due to the increment of the dimension is very low as this method requires polynomial time.

The final objective of this work is to help administrative staff of public schools in Spain, where this task is usually performed manually.

The paper is organized as follows. Section 2 describes the specific timetabling problem we are dealing with. In Section 3, the proposed modifications to the evolutionary operators of the Genetic Algorithms are presented. Simulation results are shown and discussed in Section 4. Finally, conclusions are summarized in the last section.

2 The Timetabling Problem

The timetabling problem consists of fixing a sequence of meetings between teachers and students in a prefixed period of time, typically a week. For the purpose of this paper, a simple problem of finding the optimal timetable for one academic year is considered. It works with a certain number N_g of classes, each one representing a group of students taking an identical set of subjects and usually, staying together all

the week long. There will be certain number of matters to be imparted, N_a , that will be the same for every class, and each subject will have a different number of hours per week $N_h(a)$, where $a \in [1, N_a]$. The total number of hours per week will be:

$$N_T = \sum N_h(a) . \quad (1)$$

The timetable is split in time slots, for example, an hour or fifty minutes. The teaching of each subject has to be fitted to that slot, and the arrangement of these items will define each configuration Figure 1. The number of items per day can be specified for each particular situation, as this will affect the fitness value. The number of available teachers per subject is also needed be defined. If this number, N_p , is more than one, each of the available teachers will be assigned to each class.

Maths	Sports	Literature	Sports	Maths
Literature	Music	Maths	Physics	Music
Music	Biology	Spanish	Chemistry	Literature
Literature	Maths	Physics	Music	Music
Chemistry	Literature	Maths	Literature	Physics
Maths	Maths	Physics	Music	Chemistry
Music	Spanish	Biology	Maths	Biology
Biology	Sports	Sports	Spanish	Maths

Fig. 1. An example of a possible timetable

There is a basic constraint that directs the searching: the feasibility of the timetable. There can not be more neither less hours of each subject than required, the same for teachers, etc. This constrain is included in the formulation of the problem, and the GA routines have been adapted to restrict the search space only to these group of feasible solutions. The rest of the constrains are included in the fitness function and are the ones that determine the kindness of the possible configurations.

3 The Modified Genetic Algorithms Approach

GA are based in the principles of evolution [12]. They work with a set of individuals that represent the possible solutions of the problem, and they evolve over a number of generations through reproduction and mutation techniques, exploring the solution space and moving towards the most promising areas to approach to the best individual, the one that maximizes or minimizes the fitness function. At the same time, this computational iterative method keeps diversity as different solutions.

The basic algorithm has been adapted to the problem under consideration in order to limit the search to the feasible solution space. The main modifications introduced in the standard GA concern:

- The modeling of the problem, using n -dimensional chromosomes. Each individual is represented with a matrix (chromosome) that contains as many genes as the number of hours in a week of the timetable. Usually these genes are independent being selected randomly, but in this case the interrelationship between genes will determine the feasibility of the solution.

- Generation of the population, less random than usual and restricted to the problem frame to remain in the feasible solutions space.
- Reproduction and mutation operators have been modified in order to improve its performance.

3.1 Codification of the Problem

Each individual of the population contains all the information relative to a feasible timetable. The chromosome structure is a matrix in which the rows identify the group (class 1 to class N_g), and the columns represent each time item (*gene*), arranged in increasingly order from first hour Monday morning to last hour Friday afternoon. Each subject of the set $[1..N_a]$ is codified with a number (Maths $\leftarrow 1$; Biology $\leftarrow 2$; Literature $\leftarrow 3$, ...) Figure 2.

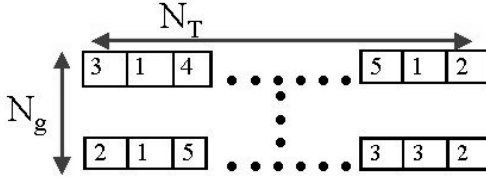


Fig. 2. Representation of a timetable as a chromosome

3.2 Population Generation

There are different possible approaches to generate individuals of the population. We have N_T slots in each group, and N_g groups, and they have to be filled with elements from N_a . If the number of chosen elements of each type is not limited, i.e., if each element was chosen randomly from $[1 .. N_a]$, the dimension of the solutions space is:

$$V = (N_a)^{N_T \cdot N_g} \quad (2)$$

Considering $N_T = 20$ hours per week, $N_a = 5$ different subjects, and $N_g = 4$ groups, there would be $V = 8.10^{55}$ possible configurations.

A particular generation function has been defined to reduce this search space, limiting the number of elements of each type according to their availability $N_h(a)$. Therefore we guarantee the feasibility of the generated chromosomes, as they will not have more elements of a specific type than allowed. The dimension of the search space in this case is reduced to:

$$V = N_T! / \prod N_h(a)! \quad (3)$$

when the selection of one element depends on the previous element selected.

Considering the same example as before, if the distribution of the $N_T = 20$ hours is $[3, 3, 4, 4, 6]$ in the 5 courses (N_a), i.e., 3 hours of N_{a_1} + 3 hours of N_{a_2} + 4 hours of N_{a_3} , ..., the search space is reduced to $V = 7.10^{44}$, that is 10^{11} times smaller, where N_{a_1} means Maths, etc.

The implementation of this method is similar to that of an innocent hand extracting balls from a bag, Figure 3. In that “bag” the assigned codes of every subject are stored, but they can be repeated as many times as hours this course is going to be taught in a week. In this example, there will be 4 hours of subject 1, 3 hours of subject 2, etc. Choosing randomly each of these genes, it is possible to find different timetables for each class and each individual. As subject 3 has been chosen, that means that first hour Monday will be Literature, see Figure 2. The numbers on the right vector are the subjects left to be arranged.



Fig. 3. Extraction of the genes to generate a possible solution

3.3 Fitness Function

The aim of the optimization is to maximize the fitness function (4) where R_i means the i -restriction, and each constraint is weighted by a value w_i which assigns the relative importance of that constraint. This weight has an influence on the evolution of the population regarding that constraint, as high weights will be accomplished sooner.

$$F_{obj} = \sum w_i R_i . \quad (4)$$

Each unfulfilled constraint is represented with a negative amount, so the optimal solution has a fitness value of zero.

Some of the requirements included in the fitness function are,

1. Avoiding teachers meeting two classes at the same time, and vice versa. This constraint is more restrictive as the number of groups increases, leading to a non-feasible solution of the problem when the number of teachers is too low.
2. It is not allowed to have a timetable with the same course twice in the same day. The item size is considered as the maximum time that one subject can be imparted per day.

Other particular constraints can be added, for example: allowing to have a particular subject at a specific hour or weekday; if two courses share the same resource they should no be placed at the same time, etc.

3.4 Reproduction and Mutation

The parents selection is performed by the roulette method. Accordingly to the value of the fitness function, each individual is assigned to a proportional sector size, which represents the probability of being selected as a progenitor of the next generation. In consequence, the best individual has the higher probability of being selected but other individuals can also be selected. This strategy allows to achieve better solutions taking into account individuals that had not been considered at that moment as the best configuration.

Once the progenitors have been selected, standard GA programs create next generation by swapping genes from two parents. Using the common crossover

operator, as the chromosomes would be split and exchanged, it is almost certain that the resulting chromosome will not be feasible due to the interdependency of the genes, Figure 4. Swapping portions of different individuals destroys the feasibility ($N_i \neq \text{cte}$).

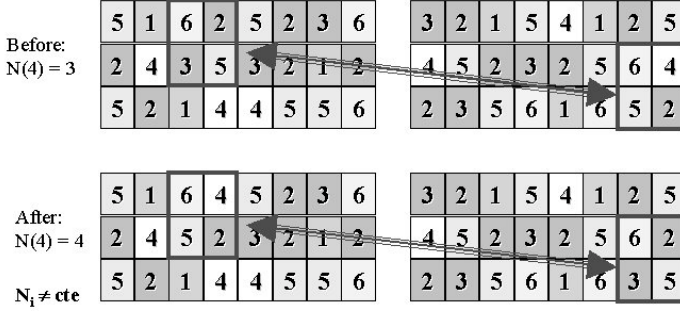


Fig. 4. Common crossover operator

In the literature, a crossover mechanism that respects this hard condition is explained in [3], or a strategy that repairs the obtained chromosomes is described [4], although these methods are quite time consuming. In this implementation the new adapted algorithm is always searching the space of feasible solutions.

Some drawback of the crossover operator are: it can violate some of the hard constraints, the necessity of repairing the springs, time consuming, etc. Those disadvantages do not appear in the mutation operation. Intrinsically, a timetable has itself all the necessary information to be an optimal solution. It needs just to be properly sorted. Hence, mixing information from different individuals is not going to improve any of those solutions, as each of them fulfills the constraints according to the disposition of the whole group of their genes.

Therefore, a parthenogenesis-like operator has been used. Each individual can produce a spring with a modified chromosome, which is generated by the reorganization of its own genes. This reorganization can be performed in a random way or with certain guidance, i.e., giving lower probability of exchange to those genes that penalise the fitness function.

In conclusion, the mutation operator selects two swapping points of the chromosome that define the the set of genes that will be exchanged. The depth of this slice (how many rows) is also a random parameter. One of these points is selected with a certain probability (P_r) from the set of conflictive genes, Figure 5.

```

If (code_repeated_same_day = true) then
    Fitness_f = Fitness_f - W;
    Conf_gen = [Conf_gen repeated_gen];
end
If (random(0..1) < Pr) then
    Swap_point = random(Conf_gen);
else
    Swap_point = random(1..number_genes);
end

```

Fig. 5. Code for implementing the mutation operator

This method helps to search better solutions, resulting in a guided influence on the generation of new solutions. The obtained chromosome will be quite similar to its parents, and most of the times better, although some times it is possible to notice some oscillations, Figure 6.

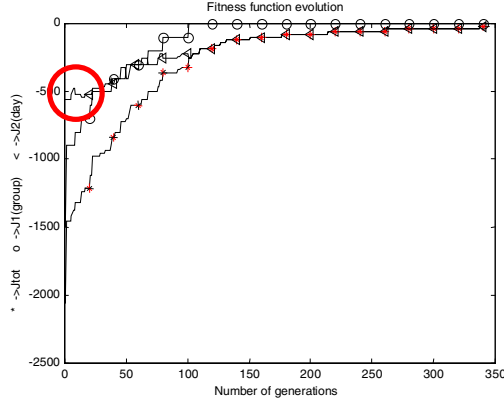


Fig. 6. Evolution of the fitness functions for the best individual in each generation

In this figure, both constraints: overlapping of subjects in different classes (<), and same subject in the same day (o), are represented, as well as the total fitness value (*). The fitness subfunction oscillation is remarked.

4 Analysis of Simulation Results

The algorithm has been implemented using Matlab, and a graphical user interface has been developed for this application, Figure 7. The interface allows us to configure the genetic algorithms parameters, such as number of substitutions, population size, Pr, etc. The maximum number of generations can be bounded. School features are also considered as configuration parameters, as number of classes, number of hours per day, number of subjects, teachers assigned to each course, etc.

Dealing with the problem of four classes, with 32 hours per week each, 10 different subjects and 19 teachers, the algorithm has used 10 individuals, 30% substitutions each generation, and $P_r = 1$, requiring less than 500 iterations to obtain the optimum.

The time penalization when increasing the population size is very high, about 1 second per additional individual. Table 1 shows how the population increase does not lead to a better solution in less generations when the population is more than 20 individuals. It is not going to be faster in any case, but it may be a softer penalization if a parallel processor computer can be used to run the algorithm. Anyway, the increment of resources is not justified by the time saved.

We have studied the increase in time when the size of the problem is on the rise. When increasing the number of classes, the number of iterations does not increase so fast as the problem dimension. Figure 8 shows the logarithm of the number of iterations (i.e., computational time) and the logarithm of the feasible search space dimension, E.

Nº individuals

10

Nº generations

100

P. cross

1

Nº substitutions

4

Nº hours/day

Monday

7

Tuesday

7

Wednesday

6

Thursday

6

Friday

6

Nº groups

4

Acceptar

Subject	Nº Hours/week	Nº teachers
Literature	4	1
History	4	2
Biology	4	1
Spanish	3	1
Physics	4	2
Maths	4	2
Chemistry	4	1
Arts	3	1
Religion	1	3
Lab	1	5

Fig. 7. Graphical User Interface developed for the application

Table 1. Iterations needed to reach the optimum according to population size

Population size	Nº iterations to optimum
2	May not find solution
4	May not find solution (>1000 iterations)
10	~500
15	~400
20	~250
30	~250
50	~250

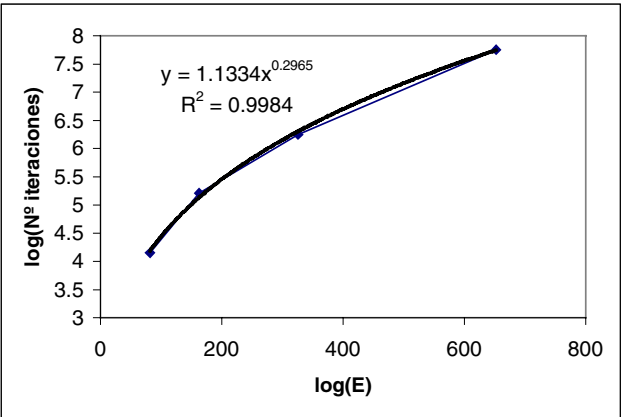


Fig. 8. Number of iterations related to the problem dimension

As it is possible to see, the curve is not linear neither exponential. The algorithm is efficient in the sense that the computational time increase is lower than linear. The fitted curve gives the following relationship, with $D = (N_a!)^{N_g}$:

$$N^\circ \text{ iterations} = 1.1334 D^{0.2965} . \quad (5)$$

Figure 9 shows how the computing time increment varies linearly with the number of generations, as it could be expected (~ 78 ms/generation). That means that the penalization due to the increment of the dimension is very low with this algorithm.

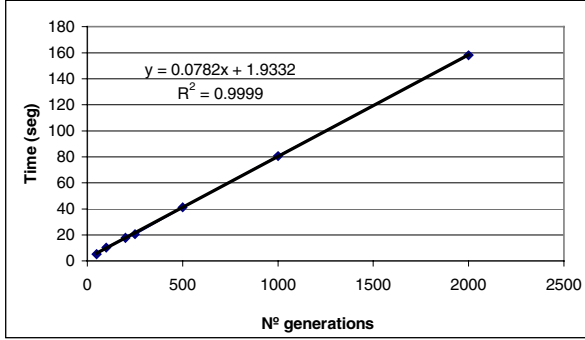


Fig. 9. Algorithm computation time vs. number of generations

One of the main keys to design the algorithm for a particular problem is the definition of the weights associated to each constraint in the fitness function. It is important to consider not only the priority of a constraint but also to remark that if a timetable violates many of the same type of soft constraints, it will have worst fitness value than another that violates a very critical one. Commissioning these weights is a crucial task for each particular problem.

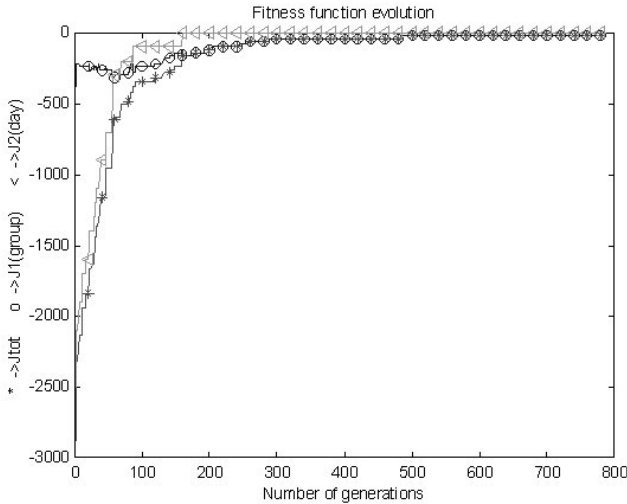


Fig. 10. Fitness function evolution with exchanged weights in fitness subfunctions

Class 1

Maths	Chemistry	Biology	Biology	Physics
Lab	History	Literature	Physics	Maths
Chemistry	Maths	Spanish	Maths	Spanish
Biology	Literature	Arts	Arts	Biology
Physics	Arts	History	Literature	Literature
History	Spanish	Chemistry	Chemistry	History
Religion	Physics			

Class 2

Biology	Literature	Maths-p.2	Maths-p.2	Biology
Chemistry	History-p.2	Spanish	Literature	Spanish
Lab-p.2	Biology	Arts	History-p.2	Arts
Literature	Physics-p.2	Literature	Chemistry	History-p.2
Religion-p.2	Spanish	Chemistry	Biology	Maths-p.2
Maths-p.2	Chemistry	History-p.2	Physics-p.2	Physics-p.2

Class 3

Spanish	Biology	History	Lab-p.3	Literature
Physics	Maths	Maths	Chemistry	Biology
Arts	Physics	Literature	History	Maths
Maths	Spanish	Spanish	Literature	Physics
Biology	Chemistry	Arts	Arts	History
Chemistry	Literature	Physics	Biology	Chemistry
Religion-p.3	History			

Class 4

Chemistry	Religion	Arts	Maths	Maths
History	Arts	History	Biology	Arts
Literature	History	Physics	Spanish	Physics
Physics	Physics	Chemistry	History	Literature
Maths	Literature	Biology	Chemistry	Spanish
Lab-p.4	Biology	Maths	Literature	Biology
Spanish	Chemistry			

Fig. 11. Timetables obtained using the modified GA

Comparing Figure 10 and Figure 6, it is worth noting how the constraint with the highest weight is the one that improves faster, while the lowest one can move towards worse values, so it will take longer to be fulfilled. The weights used here were -100 and -20. When the lowest value is used for the constrain of repeated subject in the same day, the algorithm finds the optimum sooner.

When the program finds an optimal solution it shows the results as in Figure 11.

5 Conclusions

A genetic algorithm has been adapted to a particular combinatorial problem with hard constraints. Some of the usual concepts of genetic algorithms have been modified to increase its effectiveness in this application. For example, in this case the crossover operator has not been used, and the mutation operation is less random than usual.

This strategy of using a guided mutation operator allows reaching optimal solutions in fewer iterations. These iterations are lower time consuming and less demanding, as no repairs need to be made in the new generated timetables, already feasible. That is, providing some intelligence to the algorithm, the solution can improve.

In addition, it has been shown how this method reaches optimal solutions in a pretty fast way, requiring polynomial time when the dimension of the problem increases. Computational time dependence on population size and on other variables are also discussed.

The main problem is the commissioning of the weights in the fitness function, because although including as many constraints as desired is very simple, giving them the appropriate influence in the global evaluation is a delicate task that affects the problem resolution.

It is important to notice that when a hard constraint has been included, despite it has a big influence in the algorithm, it helps to reduce the search space. So, although the design requires more efforts, the computational time is reduced and the effectiveness of the algorithm is improved.

Summarizing, this method has turned out to be effective in this kind of combinatorial problems exhibiting a good compromise between resolution time and optimal reaching.

References

1. Burke, E.K., Elliman, D.G., Weare, R.F.: A Genetic Algorithm Based University Timetabling System. In: 2nd East-West Int. Conf. on Comp. Tech. in Education, Ukraine, 1 (1994) 35–40.
2. Müller, T.: Some Novel Approaches to Lecture Timetabling. In: Proc. of the 4th Workshop on Constraints programming for Decision and Control, Gliwice, Poland (2002) 31–37.
3. Carter, M.W., Laporte, G.: Recent Developments in Practical Course Timetabling. Lecture Notes in Computer Science, Vol. 1408, Springer-Verlag, Berlin Heidelberg, NY (1998) 3–19.
4. Caldeira, J.P., Rosa, A.C.: School Timetabling Using Genetic Search. In: Proceedings of the 2nd Int. Conf. on the Practice and Theory of Automated Timetabling, Toronto (1997).
5. dos Santos, A.M., Marques, E., Ochi, L.S.: Design and Implementation of a Timetable System Using a Genetic Algorithm. In: Proceedings of the 2nd Int. Conf. on the Practice and Theory of Automated Timetabling, Toronto (1997).
6. Michalewicz, Z., Fogel, D.B.: How to Solve It: Modern Heuristics. Springer Verlag, 2000.
7. Cooper T.B., Kingston, J. H.: The Complexity of Timetable Construction Problems. Lecture Notes in Computer Science, Vol. 1153, Springer-Verlag, Berlin Heidelberg, NY (1996) 283–295.

8. Bartak, R.: Dynamic Constraint Models for Planning and Scheduling. In: New Trends in Constraints, Lecture Notes in Artificial Intelligence, Vol. 1865, Springer-Verlag, Berlin Heidelberg, NY (2000) 237–255.
9. Michalewicz, Z., Janikow, C.Z.: Handling Constraints in Genetic Algorithms. In: Proc. of the 4th Int. Conf. on Genetic Algorithms, Morgan Kaufmann Publishers (1991) 151–157.
10. Meisels A., Lusternik, N.: Experiments on Networks of Employee Timetabling Problems. In: Proceedings of the 2nd Int. Conf. on the Practice and Theory of Automated Timetabling, Toronto (1997).
11. Goldberg, D.E.: Genetic Algorithm in Search, Optimization and Machine Learning. Reading, Mass. Addison-Wesley (1989).
12. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996).
13. Schaerf, A.: A Survey of Automated Timetabling. Artificial Intelligence Review, 13 (2), (1999) 87–127.

Application of Uncertain Variables to Task and Resource Distribution in Complex Computer Systems*

Z. Bubnicki

Institute of Control and Systems Engineering
Wroclaw University of Technology
Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland
bubnicki@ists.pwr.wroc.pl

Abstract. The paper is concerned with an allocation problem for a class of complex computer systems described by knowledge representations with unknown parameters (in particular, unknown execution times). The unknown parameters are assumed to be values of uncertain variables characterized by certainty distributions given by an expert. The formulation and solution of the optimal allocation problem for a set of parallel processors are presented. The extensions to cascade structure and to cases with uncertain and random parameters are discussed. A simple example illustrates the presented approach.

1 Introduction

It is well known that computational tasks allocation or scheduling in multiprocessor systems may be connected with great computational difficulties even in the case when the scheduling may be reduced to a simple assignment problem (e.g. [1,9]). On the other hand, the execution times may not be exactly known, which leads to an uncertain decision problem (e.g. [10]).

In the works [2,3,4,5] uncertain variables and their applications to decision problems in uncertain systems described by a relational knowledge representations have been presented. The uncertain variables are described by certainty distributions given by an expert and characterizing his / her knowledge concerning approximate values of unknown parameters. They may be applied to different analysis and decisions problems in uncertain control systems.

The purpose of this paper is to show how the uncertain variables may be applied to the allocation of computational tasks in a set of parallel processors. It is considered as a specific control problem for a complex of operations. We shall consider a rather simple allocation problem consisting in the distribution of a great number of elementary tasks (elementary programs or parts of programs). The problem becomes difficult if the execution times of the tasks are unknown. In Secs 3, 4, 5 we assume that the execution times are approximately described by their certainty distributions given by an expert and we show how to determine the allocation for which the requirement concerning the execution time of the global task is satisfied with the

* This work was supported by the Polish State Committee for Scientific Research under the grant no. 4 T11C 001 22.

greatest certainty index. Sec. 6 presents an extension of the resource allocation problem to multiprocessor systems with a cascade structure. A case when there are random and uncertain parameters in the descriptions of the processors is discussed in Sec.7. Sec. 2 contains a very short presentation of the uncertain variables. Details may be found in [2,3,4].

2 Uncertain Variables and Decision Problem

Let us consider a universal set Ω , $\omega \in \Omega$, a vector space $X \subset R^k$ and a function $g : \Omega \rightarrow X$. Assume that for the fixed ω the value $\bar{x} = g(\omega)$ is unknown and introduce two soft properties: The property “ $\bar{x} \cong x$ ” which means that “ $\bar{x}$ is approximately equal to x ” or “ x is the approximate value of $\bar{x}$ ”, and the property “ $\bar{x} \tilde{\in} D_x$ ” (where $D_x \subseteq X$) which means that “the approximate value of $\bar{x}$ belongs to D_x ” or “ $\bar{x}$ approximately belongs to D_x ”. For the fixed x a soft property concerning x becomes a proposition in multi-valued logic and its logic value belongs to $[0,1]$. The logic values of our properties are denoted by v and are called *certainty indexes*. The variable $\bar{x}$ is called an *uncertain variable*. Two versions of uncertain variables have been introduced.

Definition 1 (uncertain variable). The uncertain variable $\bar{x}$ is defined by the set of values X , the function $h(x) = v(\bar{x} \cong x)$ such that $\max h(x) = 1$ (i.e. the certainty index that $\bar{x} \cong x$, given by an expert) and the following definitions:

$$v(\bar{x} \tilde{\in} D_x) = \begin{cases} \max_{x \in D_x} h(x) & \text{for } D_x \neq \emptyset \\ 0 & \text{for } D_x = \emptyset, \end{cases} \quad (1)$$

$$v(\bar{x} \not\tilde{\in} D_x) = 1 - v(\bar{x} \tilde{\in} D_x),$$

$$v(\bar{x} \tilde{\in} D_1 \vee \bar{x} \tilde{\in} D_2) = \max \{v(\bar{x} \tilde{\in} D_1), v(\bar{x} \tilde{\in} D_2)\},$$

$$v(\bar{x} \tilde{\in} D_1 \wedge \bar{x} \tilde{\in} D_2) = \begin{cases} \min \{v(\bar{x} \tilde{\in} D_1), v(\bar{x} \tilde{\in} D_2)\} & \text{for } D_1 \cap D_2 \neq \emptyset \\ 0 & \text{for } D_1 \cap D_2 = \emptyset. \end{cases}$$

The function $h(x)$ is called a *certainty distribution*. □

Definition 2 (C-uncertain variable). C-uncertain variable $\bar{x}$ is defined by the set of values X , the function $h(x) = v(\bar{x} \cong x)$ given by an expert, and the following definitions:

$$v_c(\bar{x} \tilde{\in} D_x) = \frac{1}{2} [\max_{x \in D_x} h(x) + 1 - \max_{x \in X - D_x} h(x)], \quad (2)$$

$$v_c(\bar{x} \not\tilde{\in} D_x) = 1 - v_c(\bar{x} \tilde{\in} D_x),$$

$$v_c(\bar{x} \tilde{\in} D_1 \vee \bar{x} \tilde{\in} D_2) = v_c(\bar{x} \tilde{\in} D_1 \cup D_2),$$

$$v_c(\bar{x} \tilde{\in} D_1 \wedge \bar{x} \tilde{\in} D_2) = v_c(\bar{x} \tilde{\in} D_1 \cap D_2). \quad \square$$

The certainty distribution for a particular x evaluates the expert's opinion that $\bar{x} \cong x$. In the case of C-uncertain variable the expert's knowledge is used in a better way but the calculations are more complicated.

The uncertain variables may be used in the formulation and solving a decision problem for an uncertain system. Consider a static system with input vector $u \in U$ and output vector $y \in Y$, described by a relation $R(u, y; x) \subset U \times Y$ where $x \in X$ is an unknown vector parameter which is assumed to be a value of an uncertain variable $\bar{x}$ with $h_x(x)$ given by an expert.

Decision problem. For the given $R(u, y; x)$, $h_x(x)$ and D_y (where $y \in D_y$ is a desirable output property) find u^* maximizing $v(\bar{y} \tilde{\in} D_y)$. For the fixed u

$$v(\bar{y} \tilde{\in} D_y) \stackrel{\Delta}{=} v(u) = v[\bar{x} \tilde{\in} D_x(u)] = \max_{x \in D_x(u)} h_x(x) \quad (3)$$

where $D_x(u) = \{x \in X : (u, y) \in R \rightarrow y \in D_y\}$ and $u^* = \arg \max v(u)$. When $\bar{x}$ is considered as C-uncertain variable it is necessary to find v (3) and

$$v(\bar{y} \tilde{\in} Y - D_y) = v[\bar{x} \tilde{\in} X - D_x(u)] = \max_{x \in X - D_x(u)} h_x(x).$$

Then, according to (2) with y in the place of x ,

$$v_c(\bar{y} \tilde{\in} D_y) \stackrel{\Delta}{=} v_c(u) = \frac{1}{2} [v(\bar{y} \tilde{\in} D_y) + 1 - v(\bar{y} \tilde{\in} Y - D_y)] \quad (4)$$

and $u_c^* = \arg \max v_c(u)$.

3 Application of Uncertain Variables to Task Allocation in a Set of Parallel Processors

The approach presented in the previous section may be applied to a problem of the tasks allocation in the group of parallel processors with uncertain execution times. We assume that the global computational task to be distributed may be decomposed into N separate parts (programs or parts of programs) which may be executed simultaneously by the separate processors. Each partial task is characterized by an upper bound of the execution time τ_j for j -th processor ($j = 1, 2, \dots, k$), and τ_j is assumed to be the same for each partial task. The decision problem consists in the determination of the numbers of the partial tasks $n_1, n_2, \dots, n_k$ assigned to the processors taking into account the execution time $T = \max\{T_1, T_2, \dots, T_k\}$ where T_j is the execution time for j -th processor; $n_1 + n_2 + \dots + n_k = N$. If N is sufficiently

large, we can determine the decisions $u_j \in R^+$ (any positive numbers) satisfying the constraint

$$u_1 + u_2 + \dots + u_k = N \quad (5)$$

and then obtain n_j by rounding off u_j to the nearest integer. To apply the notation in the previous section let us denote $T_j \triangleq y_j$, $T \triangleq y$, $\tau_j \triangleq x_j$, $u = (u_1, u_2, \dots, u_k) \in U$ and $x = (x_1, x_2, \dots, x_k) \in X$. Then the knowledge representation of our system is as follows

$$y_j \leq x_j u_j, \quad j = 1, 2, \dots, k, \quad (6)$$

$$y = \max\{y_1, y_2, \dots, y_k\}. \quad (7)$$

We can use more general description in the form

$$y_j \leq \varphi_j(u_j, x_j), \quad j = 1, 2, \dots, k \quad (8)$$

where φ_j are the known functions, increasing with respect to u_j . Then the relation $R(u, y; x)$ is determined by the inequalities (8) and the function (7). The unknown parameters x_j are assumed to be values of uncertain variables $\bar{x}_j$ described by the certainty distributions $h_{xj}(x_j)$ given by an expert estimating the execution times for the partial tasks. The relation $R(u, y; x)$ may be called a relational knowledge representation of the set of the processors. It is completed by the certainty distributions h_{xj} . For the known values x_j the decision problem may consist in the determination of the greatest set $D_u(x)$ of the allocations $u = (u_1, u_2, \dots, u_k)$, satisfying the implication $u \in D_u(x) \rightarrow y \leq \alpha$ where α is a given number and the property $y \leq \alpha$ is required by a user. For the uncertain parameters $\bar{x}_j$ the allocation problem may be formulated as an optimization problem consisting in finding the optimal allocation u^* which maximizes the certainty index of the soft property: “ u approximately belongs to $D_u(\bar{x})$ ” or “the set of possible values y approximately belongs to $[0, \alpha]$ ” (i.e. belongs to $[0, \alpha]$ for an approximate value of $\bar{x}$). The allocation problem is then considered as a specific version of the decision problem formulated in Sec. 2.

Optimal allocation problem. For the given h_{xj} ($j \in \overline{1, k}$) and N find the allocation $u^* = (u_1^*, u_2^*, \dots, u_k^*)$ maximizing the certainty index

$$v(u) = v\{D_y(u; \bar{x}) \subseteq [0, \alpha]\} = v[\varphi(u; \bar{x}) \lesssim \alpha] \quad (9)$$

where $D_y(u; x)$ denotes the set of possible values y for the fixed u and $\varphi(u; x)$ denotes the maximum possible value y for the fixed u . From (7) and (8) it follows that the set

$D_y(u; x)$ is described by the inequality

$$y \leq \max_j \varphi_j(x_j, u_j).$$

According to (9)

$$v(u) = v\{[\varphi_1(u_1, \bar{x}_1) \lesssim \alpha] \wedge [\varphi_2(u_2, \bar{x}_2) \lesssim \alpha] \wedge \dots \wedge [\varphi_k(u_k, \bar{x}_k) \lesssim \alpha]\}.$$

Then

$$u^* = \arg \max_u \min_j v_j(u_j)$$

where

$$v_j(u_j) = v[\varphi_j(u_j, \bar{x}_j) \lesssim \alpha] = \max_{x_j \in D_{xj}(u_j)} h_{xj}(x_j) \quad (10)$$

and $D_{xj}(u_j) = \{x_j \in R^+ : \varphi_j(u_j, x_j) \leq \alpha\}$.

The procedure of the determination of u^* is then the following:

1. To determine $v_j(u_j)$ according to (10).
2. To determine $u_1^*, u_2^*, \dots, u_k^*$ by solving the maximization problem

$$\max_{u_1, \dots, u_k} \min\{v_1(u_1), \dots, v_k(u_k)\} \quad (11)$$

subject to constraints $u_1 + u_2 + \dots + u_k = N$, $u_j \geq 0$, $j \in \overline{1, k}$. The structure of the knowledge-based decision system is presented in Fig.1. If α is sufficiently large then it is possible to obtain the allocation $u_1^*, u_2^*, \dots, u_k^*$ for which the requirement $T \leq \alpha$ is approximately satisfied with the certainty index $v = 1$.

If $\bar{x}$ is considered as C-uncertain variable then according to (4)

$$v_c[T(u, \bar{x}) \lesssim \alpha] = \frac{1}{2}[v(u) + 1 - \hat{v}(u)] \triangleq v_c(u)$$

where

$$\hat{v}(u) = v[T(u, \bar{x}) \gtrsim \alpha] = \max_j \hat{v}_j(u_j),$$

$$\hat{v}_j(u_j) = v[T_j(u_j, \bar{x}_j) \gtrsim \alpha] = \max_{x_j \in \bar{D}_{xj}(u_j)} h_{xj}(x_j)$$

and $\bar{D}_{xj}(u_j)$ denotes the complement of $D_{xj}(u_j)$.

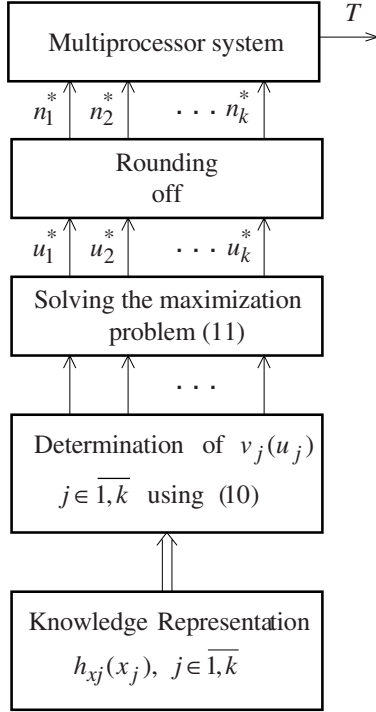


Fig. 1. Structure of the knowledge-based control system under consideration

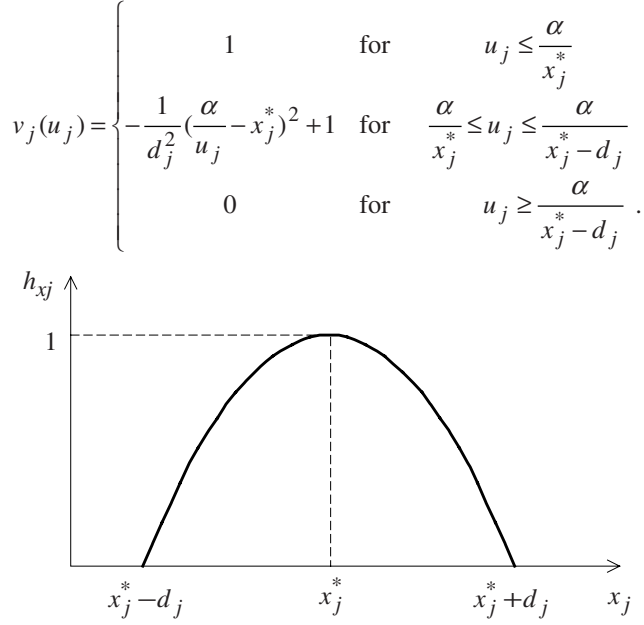
Consequently, the optimal allocation u_c^* maximizing the certainty index v_c is as follows

$$u_c^* = \arg \max_u [\min_j v_j(u_j) - \max_j \hat{v}_j(u_j)].$$

It is worth noting that in the determination of u^* the shapes of the functions $h_{xj}(x_j)$ in $\bar{D}_{xj}$ are neglected and in the determination of u_c^* the whole functions $h_{xj}(x_j)$ are taking into account.

4 Example

Assume that $T_j \leq x_j u_j$ and $h_{xj}(x_j)$ has a parabolic form presented in Fig. 2. Using (10) we obtain:

**Fig. 2.** Parabolic certainty distribution

For two processors the results are as follows:

1. For

$$\alpha \leq \frac{N(x_1^* - d_1)(x_2^* - d_2)}{x_1^* - d_1 + x_2^* - d_2} \quad (12)$$

$v(u) = 0$ for any u_1 .

2. For

$$\frac{N(x_1^* - d_1)(x_2^* - d_2)}{x_1^* - d_1 + x_2^* - d_2} \leq \alpha \leq \frac{Nx_1^*x_2^*}{x_1^* + x_2^*}$$

u_1^* is a root of the equation

$$\frac{1}{d_1} \left(\frac{\alpha}{u_1} - x_1^* \right) = -\frac{1}{d_2} \left(\frac{\alpha}{N - u_1} - x_2^* \right) \quad (13)$$

satisfying the condition

$$\frac{\alpha}{x_1^*} \leq u_1^* \leq \frac{\alpha}{x_1^* - d_1}$$

and $v(u^*) = v_1(u_1^*)$.

3. For

$$\alpha \geq \frac{Nx_1^*x_2^*}{x_1^* + x_2^*}$$

$v(u^*) = 1$ for any u_1 satisfying the condition

$$N - \frac{\alpha}{x_2^*} \leq u_1 \leq \frac{\alpha}{x_1^*}.$$

In the case (12) α is too small (the requirement is too strong) and it is not possible to find the allocation for which $v(u)$ is greater than 0. The considerations are much simpler under the assumption

$$\frac{x_1^*}{d_1} = \frac{x_2^*}{d_2} \triangleq \gamma.$$

Then the equation (13) becomes

$$\frac{\alpha}{u_1 x_1^*} + \frac{\alpha}{(N - u_1) x_2^*} = 2 \quad (14)$$

and

$$v(u^*) \triangleq v^* = 1 - \gamma^2 \left(\frac{\alpha}{x_1 u_1^*} - 1 \right)^2. \quad (15)$$

For example, if $N = 40$, $\alpha = 30$, $x_1^* = 1.5$, $x_2^* = 2$, $\gamma = 1.5$ then using (14) and (15) we obtain $u_1^* = 28$, $u_2^* = 12$ and $v^* = 0.8$ what means that 28 partial tasks should be assigned to the first processor and 12 should be assigned to the second processor, and the requirement concerning the execution time will be satisfied with the certainty index 0.8. The relationship $v^*(x_1^*, \gamma)$ for $x_2^* = 2$ (Fig. 3) shows that the parameters x_1^* , γ given by the expert may have a significant influence on the value v^* .

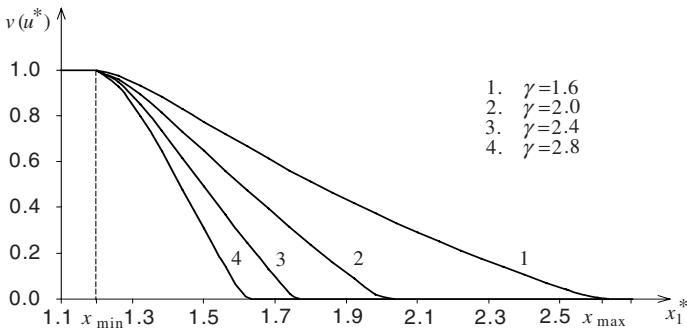


Fig. 3. Relationships between v^* and x_1^* for fixed γ

5 Resource Allocation in a Set of Parallel Processors

The similar approach may be applied to the problem of a resource allocation in the set of parallel computers. Now u_j denotes the amount of the resource assigned to the j -th processor and φ_j in (8) are decreasing functions of u_j . In particular $T_j \leq x_j u_j^{-1}$. The allocation $u = (u_1, u_2, \dots, u_k)$ should satisfy the constraint

$$\left(\bigwedge_j u_j \geq 0 \right) \wedge \sum_{j=1}^k u_j = U \quad (16)$$

where U is a global amount of the resource to be distributed. The problem consists in finding the allocation u^* satisfying the constraint (16) and maximizing the certainty index

$$v(u) = \min_j v_j(u_j) \quad (17)$$

where $v_j(u_j)$ is determined by the equation (10). The procedure of the determination of u^* is the same as in Sec. 3.

6 Resource Allocation in Multiprocessor System with Cascade Structure

Let us consider the resource allocation in a multiprocessor system with cascade structure in which the result obtained from the j -th processor is put at the input of the $(j+1)$ -th processor ($j = 1, 2, \dots, k-1$). The processors are described by the inequalities (8) and the unknown parameters x_j are described by certainty distributions h_{xj} , the same as in Sec. 3. The **optimal allocation problem** may now be formulated as follows:

For the given h_{xj} and U find the allocation $u^* = (u_1^*, u_2^*, \dots, u_k^*)$ maximizing the certainty index

$$v(u) = v\{D_y(u; \bar{x}) \subseteq [0, \alpha]\} = v[y(u; \bar{x}) \lesssim \alpha] \quad (18)$$

where

$$y(u; x) = \varphi_1(u_1, x_1) + \varphi_2(u_2, x_2) + \dots + \varphi_k(u_k, x_k)$$

denotes the maximum possible value of the execution time $T = T_1 + T_2 + \dots + T_k$.

The determination and maximization of $v(u)$ may be very complicated. It is much easier to solve the following problem: Find the allocation $\hat{u} = (\hat{u}_1, \hat{u}_2, \dots, \hat{u}_k)$ satisfying the constraint (16) and maximizing the certainty index

$$\hat{v}(u) = \max_{\alpha_1, \dots, \alpha_k} v\{[\varphi_1(u_1, \bar{x}_1) \lesssim \alpha_1] \wedge \dots \wedge [\varphi_k(u_k, \bar{x}_k) \lesssim \alpha_k]\}, \quad (19)$$

subject to constraint $\alpha_1 + \alpha_2 + \dots + \alpha_k = \alpha$. It is easy to note that $\hat{v}(u) \leq v(u)$. Then $\hat{u}$ is the allocation maximizing the lower bound of the certainty index that the maximum possible value of the execution time T is approximately less than α . According to (19)

$$\hat{v}(u) = \max_{\alpha_1, \dots, \alpha_k} \min \hat{v}_j(u_j, \alpha_j) \quad (20)$$

where $\hat{v}_j(u_j, \alpha_j)$ is obtained in the same way as v_j in (10), with α_j in the place of α , i.e.

$$\hat{v}_j(u_j, \alpha_j) = \max_{x_j \in D_{xj}(u_j, \alpha_j)} h_{xj}(x_j)$$

where

$$D_{xj}(u_j) = \{x_j \in R^1 : \varphi_j(u_j, x_j) \leq \alpha\}.$$

The presented approach may be extended to cascade-parallel structure (Fig.4).

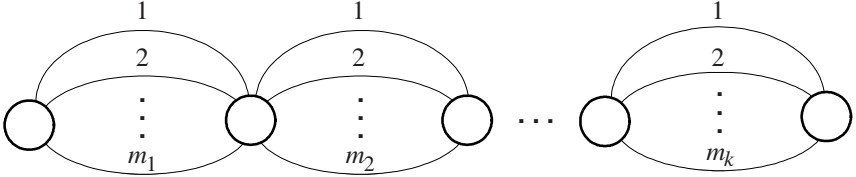


Fig. 4. Cascade-parallel structure of the multiprocessor system

The j -th subsystem ($j = 1, 2, \dots, k$) is a complex of m_j parallel processors denoted by $1, 2, \dots, m_j$. Let us denote by U_j the amount of the resource assigned to the j -th subsystem and by

$$\bar{u}_j^* = (u_{j1}^*, u_{j2}^*, \dots, u_{jm_j}^*) \triangleq f_j(U_j, \alpha_j) \quad (21)$$

the allocation in the j -th subsystem, determined in the way described in the previous section with (U_j, α_j) in the place of (U, α) , and by

$$\bar{v}_j^* = \bar{v}_j(\bar{u}_j^*) \triangleq g_j(U_j, \alpha_j) \quad (22)$$

the maximum value of the certainty index (17) for the j -th subsystem. The determination of the functions (21) and (22) may be considered as a local optimization of the subsystems. The global optimization consists in maximization of the certainty index corresponding to (20), with respect to $\alpha = (\alpha_1, \dots, \alpha_k)$ and $\bar{U} = (U_1, \dots, U_k)$, i.e. maximization

$$\max_{\bar{U}} \max_{\alpha} \min_j g_j(U_j, \alpha_j) \quad (23)$$

with constraints

$$U_1 + U_2 + \dots + U_k = U, \quad \alpha_1 + \alpha_2 + \dots + \alpha_k = \alpha.$$

Putting the results of maximization (23) into f_j in (21) we obtain the allocation $\bar{u}_j^*$ for $j = 1, 2, \dots, k$.

7 Uncertain and Random Parameters

A new problem arises when there are two unknown parameters in the description of the processor: a random parameter described by probability distribution and an uncertain parameter characterized by an expert [7]. In the place of (8), let us consider the inequality

$$y_j \leq \varphi_j(u_j, x_j, w_j), \quad j = 1, 2, \dots, k$$

where x_j is a value of uncertain variable $\bar{x}_j$ described by the certainty distribution $h_{x_j}(x_j)$ given by an expert, and $w_j \in R^1$ is a value of $\tilde{w}_j$ described by probability density $f_{w_j}(w_j)$. Consequently, the certainty index (9) and the allocation u^* maximizing this certainty index depend on w : then we obtain $v(u, w)$ and $u^*(w)$. In this case two versions of the allocation problem may be formulated and solved.

Version I. One should determine the allocation u_I maximizing the expected value of $v(u, w)$, i.e.

$$u_I = \arg \max_u E[v(u, \tilde{w})] = \int_W v(u, w) f_w(w) dw$$

where $w = (w_1, w_2, \dots, w_k) \in W$ and $f_w(w)$ is the joint probability density of $\tilde{w}_1, \tilde{w}_2, \dots, \tilde{w}_k$.

Version II. One should find the allocation u_{II} as an expected value of $u^*(w)$, i.e.

$$u_{II} = E[u^*(\tilde{w})] = \int_W u^*(w) f_w(w) dw.$$

In general, the results of these two approaches are different (i.e. $u_I \neq u_{II}$), and have different practical interpretations.

8 Conclusions

The uncertain variables are proved to be a convenient tool for solving the allocation problem in a multiprocessor system with unknown parameters in the relationship between a size of the task (or an amount of the resource) and the execution time. The similar approach may be applied to other examples of an allocation problem in the complex of operations, e.g. to manufacturing systems or to a project management under uncertainty [6,8].

Further researches may concern the application of the formalism based on uncertain variables to more complicated structures of the multiprocessor system, and the application of a learning process consisting in *step by step* knowledge validation and updating [4,6].

References

1. Blazewicz, J., Echer, K.H., Pesch, E., Schmidt, G., Weglarz, J.: Scheduling Computer and Manufacturing Processes. Springer, Berlin (1996)
2. Bubnicki, Z.: Uncertain variables and their applications for a class of uncertain systems. *International Journal of Systems Science* **5** (2001) 651–659
3. Bubnicki, Z.: Uncertain variables and their application to decision making. *IEEE Trans. on SMC, Part A: Systems and Humans* **6** (2001) 587–596
4. Bubnicki, Z.: *Uncertain Logics, Variables and Systems*. Springer-Verlag, London Berlin (2002)
5. Bubnicki, Z.: Uncertain variables and their applications for control systems. *Kybernetes* **9/10** (2002) 1260–1273
6. Bubnicki, Z.: Learning process in a class of computer integrated manufacturing systems with parametric uncertainties. *Journal of Intelligent Manufacturing* **6** (2002) 409–415
7. Bubnicki, Z.: Application of uncertain variables in a class of control systems with uncertain and random parameters. In: *Proc. of European Control Conference*. Cambridge, UK (2003)
8. Bubnicki, Z.: Application of uncertain variables to a project management under uncertainty. In: *Proc. of 16th International Conference on Systems Engineering*. Coventry, UK (2003)
9. Hsu, T. et al.: Task allocation on a network of processors. *IEEE Trans. on Computers* **12** (2000) 1339–1353
10. Tongsima, S. et al.: Probabilistic loop scheduling for applications with uncertain execution time. *IEEE Trans. on Computers* **1** (2000) 6–79

A Framework for Modelling the User Interaction with a Complex System

M^a L. Rodríguez Almendros, M^a J. Rodríguez Fórtiz, and M. Gea Megías

GEDES Group, Granada University (SPAIN)
{mlra,mjfortiz,mgea}@ugr.es

Abstract. Nowadays, the increasing performance of modern computers enables interactive systems focus on the graphical handling of information. This increase in computational power allows the designer to develop better interfaces oriented to more intuitive human-computer interaction using an interaction style more adapted to the user and system characteristic. However, when the interaction is more intuitive for the user, the design and implementation phase requires great effort by the designer. In this paper, we propose a suitable framework to analyse and design these complex systems, and a formal model that allows us to prove the system properties and validates the specification.

1 Introduction

The increasing performance of modern computers enables interactive systems focus on the graphical handling of complex systems. This increase in computational power allows the designer to develop better interfaces oriented to more intuitive human-computer interaction. Interaction style should be adapted to the user (skills, age, education, necessities, etc.) and system characteristics (components, functionality, etc.).

Interactive systems based on the desktop metaphor have adopted direct manipulation as an interaction style. This style proposes a visual model for the underlying information based on objects and actions by using input devices [1]. An interactive system based on a direct manipulation style is characterised by a user-friendly interface offering a natural representation of objects and actions [2]. This style is characterised as follows:

- Continuous representation of objects. Application objects have a graphic representation, which is visible to the users. This graphic appearance represents the object interface, and it provides information about its state (selectable, visible, active, etc.).
- Physical actions over objects. The user directly manages these objects by using input devices such as the mouse or keyboard. In this style, pointing and clicking replace the writing. Each object has certain handling capabilities (movement, dragging, dropping, etc.) that can be performed by using these input devices.
- The semantics of the action is context sensitive. An action could be carried out on several objects. Depending on the kind of object, the type of action and the gesture, the meaning of the action could be different. For example, the meaning of the

- action "dragging and dropping" could be "copy", "erase" or "print", depending on the selected object and the location where it has been dropped (a folder, the trash bin, a printer).

Direct manipulation depends on visual representation of the objects and actions of interest, physical actions or pointing instead of complex syntax, and rapid incremental reversible operations whose effect on the object of interest is immediately visible. This strategy can lead to interactive systems that are comprehensible, predictable and controllable [3]. These features represent design constraints (e.g., in task analysis) imposed by the interaction style. A formalisation of such concepts may help the designer to understand the possibilities of such an interaction style in order to achieve better applications. The direct manipulation style has traditionally been identified with the desktop metaphor environment with drag-and-drop facilities. But other approaches such as Virtual Reality, Augmented Reality or Ubiquity Computing are based on the Direct Manipulation Style, changing the space and rules by which the object of interest is shown (input and output of the traditional computer) and managed (grasping and dropping virtual objects).

These complex systems require complex graphical applications which implement:

- The interaction using new input devices and sensors to collect the user information.
- Modelling of several kinds of objects with different behaviour, features and properties.
- Management of complex interactions between objects based on spatial relationships, comparison of features, hierarchies (*part-of* or *is-a* relationships) and the way of user interaction.
- High level formal specification techniques and languages to facilitate the design of user manipulation. These techniques should also be used to analyse and verify the properties of the system in each moment.

We propose a framework [4], [5], [6] of interactive systems based on a direct manipulation style focusing on relevant aspects of this interaction model from the user's point of view. This approach allows us to describe formally the system objects, their relationships and the interaction (the user's manipulation process) with them. Our model allows us to analyse usability principles of the interactive systems based on a direct manipulation style. It also defines and checks properties of usability that every system must possess, which are deduced from the formal representation of the model. We have also design a specification method, a language, and a prototype based on this model that allow the developer to define and test the elements of the manipulation system, the relationships and the properties.

Next section will present a formal model of interactive system based on a direct manipulation style, focusing on objects and their relationships, graphical representation, interaction facilities and formal properties of this kind of interactive systems. We will propose a specification method and will present a formal language for specify the manipulation process of an user on a complex system in section 3. We will finish with conclusions section.

2 A Formal Model Based on a Direct Manipulation Style for Complex Systems

The formalisation must take into account the following elements:

- **Objects:** Taxonomy which represents the system components with handling capabilities. The system is a set of objects.
- **Functionality:** The objects carry out actions dynamically according to the user interaction.
- **Relationships:** Sets of associations between objects due to the user interaction, represent the degree of freedom for objects handling.
- **Manipulative processes:** Describe the system actions through manipulation of the objects.

2.1 Objects and System

The objects can be characterised by the following aspects:

- **Features.** The features describe the relevant characteristics of an object. They depend on the object domain. Objects have spatial features (position, orientation, etc.), visual ones (colour, appearance, etc.), modes (selected, dragged, etc.), etc. Each object has a set of features whose values determine the *state* of the object at a given time. The change in features may represent a change in state.
- **Physical representation.** Each object has a visual representation, by which it defines its external behaviour. This graphical representation constitutes the object interface and it gives information to the user about the underlying object and its internal state.
- **Actions.** Each object has functionality. Some actions are directly attached to the user, who manipulates the object by using input devices (mouse, keyboard, datagloves, etc.). These user actions change the object features (size, position, etc.). Actions can be also triggered by the object itself or by another object.

The application is composed of objects. The concepts of object and object domain are defined as follows:

Definition 1: An *object* is an element of an application with identity, features and processing capability. The object domain $\mathcal{O}$ is the set of all these elements.

Thus, we also consider the input devices as part of the object domain denoting the available object set for user handling. Therefore, these devices are included in the object domain as a subset of the latter called the *control object set* (Δ).

The internal state of an object is given by its relevant features. Therefore, we represent the set of features for the i -object, P_i . A function is defined for each one the features:

Definition 2: The *feature function* $p_{i,k}$, describes the current values of k -property for the i -object.

The values can be integers, characters, booleans, enumerate sets, etc. Each feature of set P_i has a value, and these values determine the object state. This relationship is defined by the following function:

Definition 3: The *state-feature function*, $f_{i,p}$, for an object describes the state p for the object i based on its features.

Every object has a state at any given moment. We have to know the object state at each instant of time, and therefore, the following function is defined:

Definition 4: The *state function at the instant t* , e_p , describes the object state of each object i at the instant of time t .

The system (as a whole) has a state at any given moment (describing its behaviour). This state is composed of the states of each one of its objects. Therefore, it is defined as follows:

Definition 5: The *system states* are constituted of all the possible states of the system.

Definition 6: An *image*, $\omega_{i,j}$, is a visual representation of the i -object in the j -state.

As commented above, every object has an associated visual representation. This representation allows us to handle the object, and it gives us information about its internal state. The image domain Ω defines the set of elements belonging to the system interface. To avoid ambiguity, for each element in the O set, there should be only one related visual representation or image.

The relationships between objects and their visual representation can be described with the following mapping:

Definition 7: The *representation function*, ρ , is a mapping function from objects to images.

This kind of relationship between an object and its visual representation is also known as *feedback*, that is, the observable effect of an object through the display of an interactive system. But in this context, the user not only "sees" the objects, but in fact, can interact with them. Each interface object can be freely handled, and these changes in the interface (domain) produce changes in the underlying system. Each object has a related functionality within the system. In other words, the objects provide a set of available actions that may be performed on them. When an action is performed on a set of objects, this implies a state change on the objects.

The objects of the system have handling capabilities. The user interacts with the system by carrying out actions on the objects. When the user performs an action on a set of objects, the action implies a state change on the objects, and therefore a state change on the system. This concept is formalised in the following definition:

Definition 8: An *action*, α_n , on the system is a function which changes the system state.

We denote as A the set of actions over the system. The set of actions on the system determines the system's functionality, that is, the actions that the user can carry out on the system.

The direct manipulation style represents an alternative method to access the system functionality. The user directly handles the interface objects (icons or images) to perform the user task. Therefore, the handling performed on the interface objects implies a change in their features (such as change of position, colour, size, etc), and also in their state. Therefore, we identify these processes as a special kind of actions on the system, and we describe them as gestures.

Gestures are performed by using input devices (control objects). In this process, several gestures (or devices) may be applied to obtain the same result. Therefore,

more than one gesture may be attached to a common action. For example, several gestures are provided for closing a window (pointing and clicking on the close option, typing a control sequence, etc.). Thus, for any given system action, more than one gesture may exist, and this property is expressed as follows:

Definition 9: A *gesture*, $\zeta_{n,g}$, over the system for a system action is a function defined in the image domain expressing a handling action on the interface.

The above notation for a gesture, $\zeta_{n,g}$, means that a gesture identified as g , is related to the system action n , and more than one gesture may exist for such an action. The gesture domain is denoted by ζ , the set of gestures over the system. Note that some of these gestures may be as complex as we wish. For example, dragging an image is a selection task, which involves a mouse movement, focussing on the image, pressing the mouse button, moving it and releasing the button. These interaction activities represent a change of the image location, and thus, may be denoted as a single gesture, ζ_{drag} .

The occurrence of a gesture implies a change in the objects themselves. The user has changed its visual representation, and therefore, the objects have been involved in a transformation action. By changing the interface objects we change the domain objects. The interface objects constitute the way of performing actions on the system, and therefore this feature characterises the direct manipulation paradigm. Therefore any gesture related to a system action. We may explain this relationship by the following mapping function, which relates gestures to system actions:

Definition 10: The *manipulation function*, μ , defines a mapping from the gesture domain in the actions domain.

A direct manipulation style allows the user to perform an action in different ways. Therefore, more than one gesture may be related to a certain action. The inverse is not true, however; a gesture can only perform one action at any given time.

2.2 Direct Manipulation System

As a consequence of the previous definitions we may characterise such system as follows:

Definition 11: A *direct-manipulation system* S_{md} is defined as:

$$S_{md} = \langle O, \Omega, e_t, \rho, A, \zeta, \mu \rangle$$

where O represents the object domain, Ω is the image domain, e_t is a function representing the system state in the instant of time t , ρ is the relationship between objects and their representations, A and ζ are functions representing the system actions and gestures respectively, and μ is a function mapping gestures with respect to actions.

2.3 Properties of a Direct Manipulation System

Our model allows us to represent the formal properties of this kind of interactive system. For example, it may be used to analyse usability principles [7], [8], [9], [10].

The set of properties makes it easy to achieve specified goals effectively and efficiently while promoting user satisfaction. Some such goals are:

- **Observability.** Visibility of the system state. The user should know the system state from the feedback.
- **Matching of system and real world** (use the same facts and concepts...)
- **Flexibility.** The user may have more than one way of doing something
- **Consistency.** The user should be able to predict the system response in similar situations.
- **Predictability.** The user's knowledge of the interactive history is sufficient to determine the result of the next interactions.
- **Coherence.** Any change performed in the interface is immediately applied to the corresponding objects.
- **Transparent.** Any system behaviour is observable from the user interface.
- **Fully accessible.** Any system action is accessible by the user interface using gestures.

Some of these properties can be verified within the direct manipulation system.

Definition 12: A direct-manipulation system is *consistent* if the objects features identify only one state at any time.

The consistency property is closely related to unambiguity. Therefore, it is important to guarantee that the system state is directly obtained from the object states (by the feature values) at any time.

Definition 13: A direct-manipulation system is *predictable* if each system state associates only one image for each object.

It means that the user's knowledge of the interactive history is sufficient to determine the result of the next interactions. The user determines the system state by observing the image of its objects, which represent the object state at each moment. In this way, when the user interacts with the objects, he or she can predict the following state without ambiguity. Note that the correct choice of the graphical metaphor is essential for recognition and recall.

Definition 14: A direct-manipulation system is *coherent* if any change performed in the interface is immediately applied to the corresponding objects.

An interactive system should always keep the user informed about what is going on, through appropriate feedback within a reasonable time. This property ensures that the interface handling effectively allows the user to have access to the system functionality. Therefore, a gesture implies a change in the set of objects deduced from a change in their images.

Definition 15: A direct manipulation system is *transparent* if any system behaviour is observable from the user interface.

Observability allows the user to evaluate the internal state of the system by means of its perceivable representation from the interface. This property is important because it means that the system is not viewed as a black box, and therefore, the user interface allows the user to change any feature in the object domain.

Definition 16: A direct-manipulation system is *fully accessible* if at least one gesture associated with each system action actually exists. It is related to the multiplicity of ways the user and the system may exchange information.

These properties guarantee some desirable properties of an interactive system based on a direct manipulation style. For example, it is easier to learn (learnability) and use (efficiency) if gestures are well defined. Moreover, these conditions could be imposed in the design phase as a necessary condition to achieve system usability. This theoretical analysis could help the designer in the creation of the user interface and its metaphors, guiding design decisions and prototype development. These principles can be expressed as follows:

Definition 17: A direct-manipulation system has *the property of usability* if the above properties are satisfied, that is, it is consistent, predictable, coherent, transparent and fully accessible.

Note that these properties improve system usability, but this is not a sufficient condition. Formal methods aid in the design phase, but may not ensure system usability because they are directly related to the user's evaluation. However, the formalisation may help to enhance relevant properties directly related to the user's point of view.

2.4 Study of Relationships

In a direct manipulation system the user has access to the system functionality (system actions) handling the objects. These handling processes (gestures) involve relationships among the objects of the system.

The gestures represent the syntax for direct manipulation actions and the relationships represent the semantic for these actions. Gestures produce new relationships among the objects, for example a new localisation of the objects into the space (be inside of, out of, etc.), or new values of features (an object changes the colour when it is selected, etc.). We define the following types of relationships: topological, gestures and property.

Definition 18: A *topological relationship*, $\mathcal{R}_{topological}$, is a relation between images given by their spatial location.

The set of possible relationships that could appear is close related to the possible spatial distribution of objects.

Gestures can be treated as relationships between images and control objects.

Definition 19: A *gesture relationship*, $\mathcal{R}_{gesture}$, is a binary relationship between the control object domain (Δ) and the images that gives information about whether one gesture has been done with a particular device.

Analogously, properties between objects can be treated as relationships. Any common property shared by several objects can be inquired as a relationship.

Definition 20: A *property relationship*, $\mathcal{R}_{property}$, is a relationship between objects, which gives us information about whether one common property holds.

This notation allows us to express any system event as a set of relationships between objects using a high level of abstraction. This characterisation is powerful enough to treat low-level events as well as topological relationships as object

properties caused by their handling: the former by direct manipulation and the latter as consequences of the handling.

Topological relationships, gesture relationships and property relationships are defined in the object domain and the set system relationships, as follows:

Definition 21: The set of *system relationships*, $\mathcal{R}_s$, is composed of gesture relationships, topological relationships and property relationships defined in the system for the objects.

2.5 Syntactic Direct Manipulation Model

Definition 22: A *syntactic direct manipulation model*, M_{MDS} , is composed by the tuple

$$M_{MDS} = (S_{DM}, \mathcal{R}_s)$$

where S_{DM} is a direct manipulation system and $\mathcal{R}_s$ is the set of system relationships (topological, gesture and property) defined in the direct manipulation system.

The user interacts with the objects in the system through gestures, new relationships between the objects are produced, changing the state of the system due to the modifications produced in the states of objects. The direct manipulation system and the relationships between the objects change with the time.

3 Specification Methods and Language

Once a formal model based on a direct manipulation style has been provided, a specification method is supplied to obtain relevant properties of such a system and to describe the manipulation process. When a gesture is produced in the system, an action is carried out. Specifying a manipulation process means describing the objects, the relationships and when and how the system also have to be taken into account. The object relationships and their logic and temporal dependencies have to be analysed. This allows us to identify the action to be carried out in the system, for which we define a process called the manipulative process.

Definition 23: A *manipulative process*, p_m , is a set of objects and relationships (relation-formulas) which describes an interaction with the system.

The relation-formulas are used to specify when the user can carry out an action. The set of manipulative processes in the system is denoted as P_M .

Definition 24: A *relation-formula*, f , is a finite sequence of topological, gesture and feature relationships connected by operators, which represents a condition over the system state.

We have defined tree types of operators: logical, temporal and recursive. To interpret a relation-formula, the state of the system is considered. Logical and temporal interpretations of operators and formulas are based on [11].

Therefore, based on the syntactic direct manipulation model S_{MD} and the set of manipulative processes P_M , we propose a specification method, MOORE, oriented to objects, relationships and manipulative processes.

Definition 25: The *specification model of a direct manipulation system*, S_{ORE} , is defined as the tuple

$$S_{ORE} = (M_{MDS}, P_M)$$

where M_{MDS} is a syntactic direct manipulation model and P_M is the set of manipulative processes in the system.

The domain of the manipulative processes P_M represents the semantics of the system whereas the syntactic direct manipulation model M_{MDS} is the syntactic part.

First, we define the objects and the relationships, and last we describe the interaction process between the user and the system, the manipulative processes.

We have also defines a specification language, MORELES, based on objects, relationships and manipulative processes. A specification of an interactive system using this language is divided in several modules:

- One module to specify each objects domain.
- One module to specify the system.

The module which specifies the system imports the objects modules and includes information about the objects interactions from different domains and conditions to carry out them. This division in modules facilitates the specification of the complex system, provides independence and allows the reutilisation of the specification of an object domain in different systems.

Besides, each module is divided into sections as we show:

- Object domain module: This module describes the control objects, features, states, graphical representation, actions, gestures, topological and manipulative processes of the objects of this domain.

```
<object_domain>::= <domain> [<control_object_domain_name>] <features>
                   <states> <graphical_representation> [<actions>]
                   [<gesture_relationships>]
                   [<correspondence_actions_gestures>]
                   [<topological_relationships>]
                   [<manipulative_processes>]
```

- System module: The module that specifies the system defines objects domains, actions and gestures of the system, topological and properties relationships and manipulative processes.

```
<system_specification>::= <system_name>
                          <concrete_objects_from_domains>
                          <system_actions>
                          <system_gesture_relations>
                          <correspondence_system_actions_gestures>
                          <topological_relationships>
                          [<property_relationships>]
                          <manipulative_processes>
```

Moreover, we have generated a Java prototype automatically from the specification language. A class is created for each object domain. Another class implements the system functionality and structure. Java facilitates the management of graphical representations (images) and the modeling of gestures, treating them as events.

4 An Example: Set Game

In this section we provide an example from a simple case study to illustrate how the formalism, the specification method and language we have introduced are used, showing their expressiveness. Many applications are based on spatial relationships among objects. As examples are the kid games. We have selected these examples because the direct manipulation of objects by the child is crucial for his learning process, and thus, the knowledge is given by rules (ordering, associations, etc.).

The set game is an easy game, which allows us analyse the child capacity to make associations (*belong* relationships). The rule for the game is grouping together the game pieces that have a *belong* property. For example, the set game has three objects domain: mouse, dogs and doghouses. The objective of this game is to include a dog in doghouse. The specification of the game is divided in four modules, one module to specify each object domain (*mouse*, *dogs* and *doghouses*) and one module to specify the system (*set game*).

To illustrate the use of the language, only the objects domain *dogs* is defined:

```

domain dogs;
//Control objects
import mouse;
//Attributes of the objects
features position (x,y), mode (normal, selected);
//States of the objects
states      state_normal = ( position (x,y), mode (normal)),
              state_selected = ( position (x,y), mode
              (selected));
//Graphical representations and states
graphical_representation state_normal → image_normal,
                           state_selected → image_selected;
//Actions of the objects
actions
    select: state_normal → state_selected,
    move(position,mode):(state_selected((x,y),mode:selected) →
                           state_normal((c,d), mode:normal);
//Gestures relationships of the objects
gesture_relationships  $\mathcal{R}_{\text{select}}$ (dogs),  $\mathcal{R}_{\text{move}}$ (dogs);
//Gestures relationships and actions
actions_gestures select=( $\mathcal{R}_{\text{select}}$ ), move=( $\mathcal{R}_{\text{move}}$ );
//Topological relationships
topological_relationships (mouses on dogs);
//Manipulatives processes
manipulative_processes
//Manipulative process: Select a dog
 $\mathcal{R}_{\text{select}}$ (dogs):=  $\exists!$  d  $\in$  domain(dogs),  $\exists!$  r  $\in$  domain(mouses):
    (r on d)  $\wedge$   $\mathcal{R}_{\text{click}}$ (r),
//Manipulative process: Move a dog
 $\mathcal{R}_{\text{move}}$ (dogs) :=  $\exists!$  d  $\in$  domain(dogs),  $\exists!$  r  $\in$  domain(mouses):
     $\mathcal{R}_{\text{drag}}$ (r)  $\wedge$  (r on d);

```

The control object *mouse* is used to manipulate the objects *dogs*. The dogs have two feature (*position* and *mode*), two states (*normal* and *selected*) and therefore two graphical representations. The user can *select a dog* and *move a dog* (actions and gestures *select* and *move*). This domain has a topological relationships defined, *on*, with the control objects domain, *mouses*.

Two manipulative processes have been defined: *Select a dog* implies that the mouse is over the dog (topological relationships, *on*) and the mouse button is pressed (gesture relationships, *click*); *Move a dog* implies that the mouse is over the dog (topological relationships, *on*) and the mouse is dragged to a new position (gestures relationships, *drag*), and while the mouse is being dragged the mouse is over the dog (operator $S \wedge$, which means *since and*).

5 Conclusions and Future Works

A framework of specification of complex interactive systems has been presented. It allows to model the components of a direct manipulation system, the objects, and the relationships between them. In order to allow the manipulation of the objects, actions, gestures and conditions (relation-formulas) may be specified.

The framework may be used to analyse the usability properties of the interactive system: consistent, predictable, coherent and transparent. As the system is specified formally (set theory and logic are used to describe its architecture) its properties can also be verified formally, using the same formalisms.

To facilitate to the user the description of the system, its properties and the manipulation process, a specification language has been proposed. The syntax of a specification written with this language can be translated automatically to a Java prototype. It allows that the user verifies and validates the system.

Our future work is the extension of the framework to specify the manipulation of several users at the same time. It is useful to model the interaction of a group of users in the cooperative and collaborative systems. Cooperative models and concurrency are been taken into account.

References

1. Shneidermann B.: Direct Manipulation: A Step Beyond Programming Languages. IEEE Computer, 16 (1983) pp 57–69.
2. Shneiderman, B.: Designing the User Interface. Addison-Wesley Publishing Company, Second Edition, 1992.
3. Shneiderman, B.: Direct Manipulation for Comprehensible, Predictable and Controllable User Interfaces, Proceedings of the International Conference on Intelligent User Interfaces, 1997.
4. Rodríguez, M^a. L.; Gea, M.; Gutiérrez, F. L.: Towards Spatial Specification of Interactive System. EUROGRAPHICS, 99. ISSN: 1017–4656. B.Falcidiendo, J. Rossignac (EDS) Milán, 1999.
5. Rodríguez, M^a. L.; Garvía, E.; Gea, M.: A Spatial Specification Technique for Interactive System. Proceeding of Design, Specification and Verification Of Interactive System. DSVI'2000. Limerick, Irlanda, 2000.

6. Rodríguez, M^a.L.:Aproximación Metodológica Formal para el Diseño y desarrollo de Sistemas Interactivos basados en el Estilo de Manipulación Directa. (in Spanish) Phd. Thesis, University of Granada, 2002.
7. Dix, A. J.; Ginaly, J. ; Abowd, G. ; Beale, R.: Human-Computer Interactions. 2^o Edition, Prentice Hall, 1998.
8. Ivory, M.Y.; Mearst, M.A.: The state of the art in Automatic Usability Evaluation of UI. ACM Computing Surveys, 4(33), 2001.
9. Intentional Society Organism. Usability of Computer System, 1999.
10. Nielsen, J.; Philips, V.: Estimating the Relative Usability of Two Interfaces: Heuristic, Formal, and Empirical Methods Congrared. Proceeding of INTERCHI'93, 1993.
11. Gabbay, D.M.; Hodkinson, I.; Reynolds, M.: Temporal Logic. Mathematical Foundations and Computational Aspect, Volume 1. Oxford Logic Guides 28, 1995.

A Categorical Approach to NP-Hard Optimization Problems

Liara Aparecida dos Santos Leal¹, Dalcídio Moraes Claudio¹,
Laira Vieira Toscani², and Paulo Blauth Menezes²

¹ Mathematics Department
PUCRS – Porto Alegre, Brazil
{liara,dalcidio}@pucrs.br

² Computing Institute
UFRGS – Porto Alegre, Brazil
{blauth,laira}@inf.ufrgs.br
<http://www.inf.ufrgs.br/>

Abstract. Aiming at developing a theoretical framework for the formal study of NP-hard optimization problems, which is built on precise mathematical foundations, we have focused on structural properties of optimization problems related to approximative issue. From the observation that, intuitively, there are many connections among categorical concepts and structural complexity notions, in this work we present a categorical approach to cope with some questions originally studied within Computational Complexity Theory. After defining the polynomial time soluble optimization problems category OPTS and the optimization problems category OPT, a comparison mechanism between them and an approximation system to each optimization problem have been introduced, following the basic idea of *categorical shape theory*. In this direction, we consider new insights and a deeper understanding of some basic questions inside the Structural Complexity field, by an universal language.

1 Introduction

This work is motivated by the current interest in Computer Science on Approximative Algorithms Theory as a feasible alternative to those optimization problems considered intractable. After the original success in obtaining approximative algorithms to various problems, a great research effort has been devoted in trying to find a uniform structure to deal with the notion of approximability to optimization problems, under the Complexity Theory point of view. As theoreticians continue to seek more powerful methods for proving problems intractable, parallel efforts focusing on learning more about the ways in which problems are interrelated with respect to their difficulty and comparing the complexity of different combinatorial optimization problems have been an extremely active research area during the last twenty years. The different behaviour of NP-hard optimization problems with respect to their approximability properties is captured by means of the definition of approximation classes and, under the “ $P \neq$

NP” conjecture, these classes form a strict hierarchy whose levels correspond to different degrees of approximation.

Structural Complexity Theory is often concerned with the inter-relationships between complexity classes. However, it seems that an attempt of organizing all these results in a unified framework as general as possible is lacking. The aim of this paper is to make a first step in this direction. Starting from the observation that, intuitively, there are many connections among categorical concepts and structural complexity notions, in [6,7] we define two categories: the OPTS category of polynomial time soluble optimization problems and the OPT category of optimization problems. In this direction, a comparison mechanism between the OPTS and OPT categories has been introduced in [8]. It is worth to say that in so doing we were very much inspired by the point of view adopted by Rattray [12,13] in his complex systems abstract model, based on categorical shape theory due to Cordier and Porter [3].

The introduction of an appropriate abstract notion of reductibility between optimization problems allows to formally state that an optimization problem is as hard to approximate as another one. In particular, the notion of approximation-preserving reductibility orders optimization problems with respect to their difficulty of being approximated. *Hard problems* are the maximal elements in a class, with respect to this order, and capture the *essential* properties of that class. In this sense, NP-hard problems are *universal* to NPO class.

The notion of *universality* is fundamental to the category theory. According to D. Ellerman [4], “the category theory’s foundational relevance is that it provides universality concepts to characterize the important structures throughout mathematics.” In this context, the notion of *universal* for a property represents the *essential* characteristics of such a property without any imperfections, and category theory is a precise mathematical theory of *concrete universals*. The foundational role of category theory is to characterize what is important in mathematics by exhibiting its concrete universality properties.

A preliminary version of this work appeared in [10]. However as there have been some advances since that time, it has seemed advisable to present an improved version. As mentioned earlier, our aim is to present a theoretical framework for the formal study of NP-hard optimization problems, focusing on structural properties related to approximative issue. In the present paper we consolidate the categorical approach, explaining the relationships between the formal and informal notions that underline our work.

2 Basic Questions

In the present work we are interested in to provide an universal language for supporting formalisms to specify the hierarchy approximation system for an abstract NP-hard optimization problem, in a general sense. Having defined both the polynomial time soluble optimization problems category and the optimization problems category, the next step is to identify the relationships between them. We start from these basic questions, introduced in [9].

1. How do OPTS and OPT categories interact with each other?
2. What does it mean to say that a problem A “approximates” an optimization problem B?
3. What is it understood by the “best approximation” for such an optimization problem?

The goal is now to provide mechanisms for the comparison between such categories. In the scenario of categorical shape theory, we may consider the OPT category as the category of objects of interest $\mathbf{B}$, the OPTS category as the category of archetypes $\mathbf{A}$, and $K: \text{OPTS} \rightarrow \text{OPT}$ would be a comparison mechanism related to an approximation method (for instance by using relaxation). Through this theory it is possible to identify the best approximation to an optimization problem B, if it exists.

In order to characterize approximation degrees by means of categorical shape theory, the basic idea is the construction of a system approximation to each optimization problem using limits (or colimits). A limit construction provides a means of forming complex objects from patterns (diagrams) of simpler objects. By using limits, a hierarchical structure can be imposed upon the system of approximation.

Right now, we consider the new insights and a deeper understanding of the central questions and their implications. The direction is aimed towards actually exploring the connections among the structural complexity aspects and categorical concepts, which may be viewed in a “high-level”, in the sense of a structural complexity approach.

3 Mathematical Foundations

In order to make the paper self-contained, this section gives some basic categorical concepts following the literature [2], and introduces briefly the Theory of Universals, based on the paper by D. Ellerman [4].

3.1 Category Theory

According to Barr and Wells [2], there are various view on what category theory is about, and what it is good for. Category theory is a relatively young branch of mathematics stemming from algebraic topology, and designed to describe various *structural* concepts from different mathematical fields in a *uniform* way. Indeed, category theory provides a bag of concepts (and theorems about those concepts) that form an abstraction of many concrete concepts in diverse branches of mathematics, including computing science. Hence, it will come as no surprise that the concepts of category theory form an abstraction of many concepts that play a role in Structural Complexity.

Definition 1. A category $\mathbf{C}$ is specified by a collection $ob\mathbf{C}$, disjoint sets $\mathbf{C}(A,B)$ for $A, B \in ob\mathbf{C}$, and an associative operation $\circ$, such that (i) $(f \circ g)$ is defined for $g \in \mathbf{C}(A,B)$, $f \in \mathbf{C}(B,C)$ if and only if $B=C$; (ii) for each $A \in$

obC, there exists $1_A \in \mathbf{C}(A, A)$ such that $(1_A \circ f) = f$ and $(g \circ 1_A) = g$, whenever the composition is defined.

Definition 2. A functor $F: \mathbf{A} \rightarrow \mathbf{B}$ for the categories $\mathbf{A}$ and $\mathbf{B}$ maps $\text{ob}\mathbf{A}$ into $\text{ob}\mathbf{B}$ and sets $\mathbf{A}(A, B)$ into $\mathbf{B}(FA, FB)$ such that it preserves (i) units, that is, $1_{FA} = F(1_A)$, for each object of $\mathbf{A}$; (ii) composition, that is, $F(f \circ g) = (Ff \circ Fg)$, whenever $(f \circ g)$ is defined.

Definition 3. Let $\mathbf{C}$ be a category, and A any object of $\mathbf{C}$. The comma category $\mathbf{C} \downarrow A$ is the category of objects over A such that it has $\mathbf{C}$ -morphisms with codomain A as objects, and as morphisms from $f: B \rightarrow A$ to $g: C \rightarrow A$ the $\mathbf{C}$ -morphisms $k: B \rightarrow C$, where $g \circ k = f$.

3.2 Theory of Universals

The *Theory of Universals* due to D. Ellerman [4] is originally concerned to explain many of the ancient philosophical ideas about universals, such as:

1. The Platonic notion meaning that all the instances of a property have the property by virtue of participating in the universal; and
2. The notion of the universal as showing the essence of a property without any imperfections.

The notion of *universality* is fundamental to the category theory. The foundational role of category theory is to characterize what is important in mathematics by exhibiting its concrete universality properties. The concrete universal for a property represents the *essential* characteristics of the property without any imperfections, and category theory provides the concepts to stress the universal instance from among all the instances of a property. All the objects in category theory with universal mapping properties such as *limits* and *colimits* are concrete universals for universal properties. Thus the universal objects of category theory can typically be presented as the limit (or colimit) of a process of filtering out to arrive at the *essence* of the property.

Definition 4. A mathematical theory is said to be a theory of universals if it contains a binary relation μ and an equivalence relation $\approx$ so that with certain properties F there are associated entities u_F satisfying the following conditions: (i) universality: for any x , $(x \mu u_F)$ iff $F(x)$, and (ii) uniqueness: if u_F and u'_F are universals for the same F , then $u_F \approx u'_F$.

A universal u_F is said to be *abstract* if it does not participate in itself, i.e., $\sim (u_F \mu u_F)$. Alternatively, a universal u_F is *concrete* if it is self-participating, i.e., $u_F \mu u_F$.

4 Optimization Problems Categories

In this section a categorical approach to optimization problems is presented in such way that the notion of *reduction* from a problem to another one appears, naturally, in the conceptual sense of *morphism* between two objects. Reductibility provides the key-concept to this approach. The recognition that the only structure that an object has is by virtue of its interaction with other object leads to focus on structural aspects of optimization problems. A preliminary and short version of this idea appeared in [6].

The introduction of an appropriate notion of reductibility between optimization problems allows to formally state that an optimization problem is as hard to approximate as another one. In particular, the notion of approximation-preserving reductibility orders optimization problems with respect to their difficult of being approximated. *Hard problems* are the maximal elements in a class, with respect to this order, and capture the *essencial* properties of that class. In this sense, NP-hard problems are *universal* to NPO class.

We assume that the basic concepts of computational complexity theory are familiar. We are following the notation of Garey and Johnson [5], which is universally accepted, as such as the known books by Ausiello et al.[1] and Papadimitriou [11]. In the following, we briefly review the basic terminology and notation.

4.1 Optimization Problem

On the analogy of the theory of NP-completeness, it there has been more interest in studying a class of optimization problems whose feasible solutions are short and easy-to-recognize.

Definition 5. *An optimization problem $\mathbf{p} = (I, S, Opt)$ is Nondeterministic Polynomial (NPO in short) if*

1. *The set of instances I is recognizable in polynomial time;*
2. *Given an instance x of I , all the feasible solutions of x belonging to the set S_x are short, that is, a polynomial p exists such that, for any $y \in S_x$, $y \leq p(x)$. Moreover, it is decidable in polynomial time whether, for any x and for any y such that $y \leq p(x)$, $y \in S_x$.*
3. *The objective function Opt is computable in polynomial time.*

Definition 6. *The NPO class is the set of all NPO problems, the PO class is the class of NPO problems that admits a polynomial time algorithm to find their optimum solution.*

Similarly to "P=NP?" question, also it is not known whether "PO=NPO".

By means of the notion of reductibility between optimization problems it is possible to define *hardness* to NPO class.

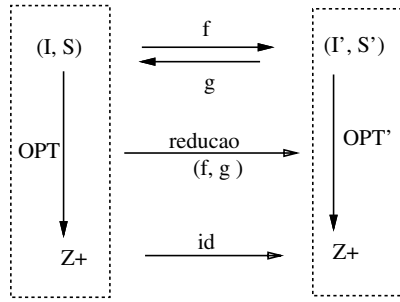


Fig. 1. Reduction between Optimization Problems

4.2 Reductions

In general, within complexity theory, a *reduction* from a problem A to a problem B specifies some procedure to solve A by means of an algorithm solving B . In the context of approximation, the reduction also should guarantee that an approximate solution of B can be used to obtain an approximate solution for A .

Definition 7. A reduction between the optimization problems $\mathbf{p} = (I, S, \text{Opt})$ and $\mathbf{q} = (I', S', \text{Opt}')$ is a pair of polynomial time computable functions (f, g) , where $f : I \rightarrow I'$ and $g : (I', S') \rightarrow (I, S)$ are such that the diagram in the figure 1 commutes.

The meaning of that diagram commutes is that, in order to obtain the optimum solution for the problem $\mathbf{p}$, it is possible firstly to reduce the problem $\mathbf{p}$ to the problem $\mathbf{q}$, and secondly to solve the problem $\mathbf{q}$. The solution obtained will be the same solution given by some procedure which solve the problem $\mathbf{p}$ directly.

Reductions are defined in such way that they are composable and they satisfy *transitivity* and *reflexivity* properties. Two problems are said *polynomially equivalent* whenever they reduce to each other. It follows that a reduction defines an equivalence relation, and thus it imposes a partial order on the resulting equivalence classes of problems.

Definition 8. Given a reduction, an NPO problem $\mathbf{p}$ is said to be NP-hard respect to that reduction, if for all NPO problems $\mathbf{p}'$ we have that $\mathbf{p}'$ reduces to $\mathbf{p}$.

It is important to observe that *hardness* means different things to different people. As a matter of convenience, the theory of NP-completeness was designed to be applied on decision problems. To other kind of problems, such as optimization problems or those problems not belonging to the NP class, should be used the hardness term. On the other hand, hard and complete problems are defined for all kind of problems. Let $\mathcal{C}$ be any class of problems and α a given reduction.

A problem p is said to be C-hard if for all problems q in C it has $q \propto p$. A C-hard problem p is said to be C-complete if in addition p belongs to C.

An approximation-preserving reduction is defined as a reduction between optimization problems adding some conditions that guarantee some property related with approximation.

Definition 9. *A approximation-preserving reduction between the NP optimization problems $\mathbf{p} = (I, S, Opt)$ and $\mathbf{q} = (I', S', Opt')$ is a triple of polynomial-time computable functions (f, g, c) , where $f : I \longrightarrow I'$, $g : (I', S') \longrightarrow (I, S)$ and $c : Z^+ \longrightarrow Z^+$ are such that the correspondent diagram commutes.*

Here is introduced a function c , which role is that of preserving the quality of approximation. Depending on the imposing relation between quality approximation of problems, several different approximation-preserving reduction have been defined in the last fifteen years [1].

4.3 OPTS Category

Definition 10. *The polynomial time soluble optimization problems category OPTS has PO optimization problems as objects and reductions between optimization problems as morphisms.*

Theorem 1. *OPTS is a category.*

Proof. Since reductions are defined as computable functions satisfying the reflexive and transitive properties, it has that identity morphisms are guaranteed by means of reflexivity, and composition with associativity is obtained by means of transitivity.

After we have given a first step in the categorical approach with the definition of the polynomial time soluble optimization problems category, it is natural to pursue in this direction, aiming at extending to NPO optimization problems considered intractable. Next, considering the notion of approximation-preserving reduction as morphisms between optimization problems, it is possible to define an wider category.

4.4 OPT Category

Definition 11. *The optimization problems category OPT has NPO optimization problems as objects and approximation-preserving reductions as morphisms.*

Analogously to OPTS category, is easily verified that OPT is really a category.

Theorem 2. *OPT is a category.*

Next, we present an important result, confirming that hard problems capture the essential properties of their classes.

Theorem 3. *NP-hard problems are concrete universals to OPT category.*

Proof. Given a reduction α , let U a NP-hard problem respect to α . We have to show that U is a concrete universal object for some participation relation μ and an equivalent relation $\approx$, according to Definition 4. Let the participating relation be: $(p \mu U)$ iff $(p \alpha U)$, where $F(p) \equiv$ “ p reduces to U ”, and the equivalence relation $\approx$ the polynomial equivalence relation defined in terms of the reduction α .

It has that, U satisfies the universality condition, by NP-hard problem definition, that is, for any NPO-problem p , $(p \mu U)$ iff $(p \alpha U)$. Also, since the reduction induces an equivalence relation, it has that if U' is also a NP-hard problem, then $U \alpha U'$ and $U' \alpha U$, that is, U is polynomially equivalent to U' . Therefore U is a concrete universal object to OPT category, which concreteness condition corresponds to the reflexivity property of reduction.

5 OPTS Category $\times$ OPT Category

Having defined both the polynomial time soluble optimization problems category and the optimization problems category, the next step is to identify the relationships between them. As we have mentioned earlier, we start from these basic questions:

1. How do OPTS and OPT categories interact with each other?
2. What does it mean to say that a problem A “approximates” an optimization problem B?
3. What is it understood by the “best approximation” for such an optimization problem?

The goal is now to provide mechanisms for the comparison between such categories. This will lead us to the categorical shape theory.

5.1 Categorical Shape Theory

Very often we wish to find a mathematical model of a structure in order to explain its properties and predict its behavior in different circumstances.

Related to the approximability issue to optimization problems, it is likely that the categorical shape theory would be such a model. It does provide a comparison mechanism to establish the meaning of an approximation system, identifying the universal properties in the category theory sense, in order to describe how an object “best approximates” another object.

This section has been motivated from previous work by Ratray [12], using the basic idea of categorical shape theory due to Cordier and Porter [3], which is that, in any approximating situation, the approximations are what encode the only information that it can analyze.

In the context of categorical shape theory, there are three basic defining elements:

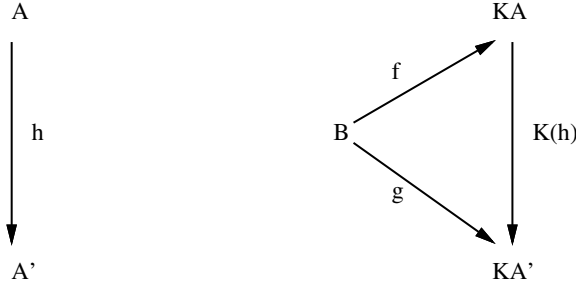


Fig. 2. Morphisms between Approximations

1. a category $\mathbf{B}$ of objects of interest;
2. a category $\mathbf{A}$ of archetypes or model-objects;
3. a “comparison” of objects with model-objects, ie. a functor $K : \mathbf{A} \rightarrow \mathbf{B}$.

The basic idea behind categorical shape theory is that recognizing and understanding an object of interest B via a comparison $K : \mathbf{A} \rightarrow \mathbf{B}$ requires the identification of the corresponding archetype A which best represents B .

Definition 12. *Given category $\mathbf{A}$ of archetypes, category $\mathbf{B}$ of objects of interest, and a comparison $K : \mathbf{A} \rightarrow \mathbf{B}$, an approximation to an object B in $\mathbf{B}$ is the pair (f, A) , where A in $\mathbf{A}$ is an archetype and $f : B \rightarrow KA$.*

A morphism between approximations $h : (f, A) \rightarrow (g, A')$ is a morphism $h : A \rightarrow A'$ of the underlying archetypes, such that $K(h) \circ f = g$, ie. the triangle in the figure 2 commutes.

According to Cordier and Porter [3], the definition of a morphism $h : (f, A) \rightarrow (g, A')$ corresponds to saying that if $g : B \rightarrow KA'$ can be written as a composite $K(i) \circ f$, where $f : B \rightarrow KA$ and $i : A \rightarrow A'$, that is

$$B \rightarrow KA \rightarrow KA'$$

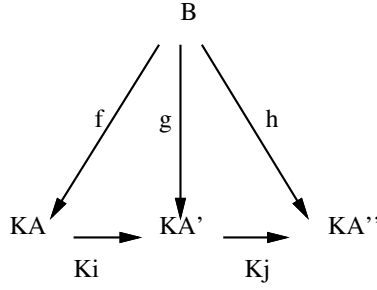
then (f, A) already contains the information encoded in (g, A') .

Thus in some way (f, A) is “finer” approximation to B than is (g, A') .

The notion of “most closely approximates” is given by a universal object.

Definition 13. *Let $K : \mathbf{A} \rightarrow \mathbf{B}$ be a comparison functor. An archetype A of $\mathbf{A}$ is said to be K -universal for an object of interest B of $\mathbf{B}$ if there exists an approximation (f, A) to B such that, for each approximation (g, A') to B , with A' in $\mathbf{A}$, there exists a unique morphism $h : A \rightarrow A'$ in $\mathbf{A}$ with $g = K(h) \circ f$.*

Approximations with their morphisms form a category $B \downarrow K$, the comma category of K -objects under B . The cone-like form of the morphisms in $\mathbf{B}$ giving the approximations for some object B , suggests that taking the limit object of the diagram would result in an archetype A^* “as near as possible “ to B . See figure 3 below.

Fig. 3. Approximations to B

Definition 14. *Category $\mathbf{A}$ is said to be K -universal in $\mathbf{B}$ if every object of interest of $\mathbf{B}$ has a K -universal archetype in $\mathbf{A}$.*

5.2 Connections with OPTS and OPT Categories

Let $\mathbf{OPT}$ be the category of objects of interest $\mathbf{B}$, $\mathbf{OPTS}$ be the category of archetypes $\mathbf{A}$, and $K: \mathbf{OPTS} \rightarrow \mathbf{OPT}$ be a comparison mechanism related to an approximation method (for instance by using relaxation).

Returning to the second question purposed in this beginning section, which is: "What does it mean to say that a problem A *approximates* an optimization problem B ?", next we define an *approximation problem*.

Definition 15. *Given a functor $K: \mathbf{OPTS} \rightarrow \mathbf{OPT}$, a problem $B \in \mathbf{OPT}$ is said an approximation problem if there are a problem $A \in \mathbf{OPTS}$ and an approximation-preserving reduction f , such that $f: B \rightarrow K A$.*

In this case, the pair (f, A) is an *approximation* to the problem B .

Through categorical shape theory is possible to identify the best approximation to an optimization problem B , if it exists. In fact, the existence of optimization problems not allowing any kind of approximation makes the proposition below consistent.

Proposition 1. *$\mathbf{OPTS}$ category is K -universal in $\mathbf{OPT}$ if and only if $PO = NPO$.*

5.3 Category of Approximations

In order to characterize approximation degrees by means of categorical shape theory, the basic idea is the construction of a system approximation to each optimization problem using colimits.

Definition 16. *Given a functor $K: \mathbf{OPTS} \rightarrow \mathbf{OPT}$, the category $APX_{B,K}$ of approximations to an optimization problem $B \in \mathbf{OPT}$ is the comma category $B \downarrow K$ of K -objects under B .*

A such kind of limit construction provides a means of forming complex objects from patterns (diagrams) of simpler objects. By using colimits in the $APX_{B,K}$ definition, a hierarchical structure can be imposed upon the system of approximation, reaching the best approximation from the system. In this context, many aspects are still being investigated and left for further works.

6 Conclusions

This work grew from the idea of providing a categorical view of structural complexity to optimization problems.

As it is known, category theory provides basic notation and a universal language for explaining, investigating and discussing concepts and constructions from different fields, in a uniform way. It allows this from a viewpoint different to that of set theory in which an object is described in terms of its “internal” structure. In categorical terms the only structure that an object has is by virtue of its interaction with other objects, described in terms of “external” features. Through categorical approach we have improved our understanding and development of many concepts within structural complexity related to approximation for optimization problems. Theory of universals has provided a mathematical foundation to explain basic universal elements from complexity theory in an elegant way.

A comparison of the OPTS and OPT categories has been motivated from previous work by C. Rattray [12], based on categorical shape theory. The study that we have started in this paper is an attempt in this direction. Along the same line, we think that in order to establish connections among optimization problems and their approximability properties, it may be fruitful to find relationships with other results drawn from other approaches, at the same level of abstraction, such as the one developed in [13]. The work is still on-going and involves many aspects of categorical shape theory. Right now, we consider the new insights and a deeper understanding of the central questions and their implications. The direction is aimed towards actually exploring the connections among the structural complexity aspects and categorical concepts, which may be viewed in a “high-level”, in the sense of a structural complexity approach.

Acknowledgments. This work was partially supported by: PUCRS – Pontificia Universidade Catolica do Rio Grande do Sul, Brazil, UFRGS – Universidade Federal do Rio Grande do Sul, Brazil and CNPq – Conselho Nacional de Desenvolvimento Cientifico e Tecnológico, Brazil.

References

1. G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela and M. Protasi, “Complexity and Approximation – Combinatorial Optimization Problems and their Approximability Properties”, Springer-Verlag, 1999.

2. M. Barr and C. Wells “Category Theory for Computing Science”, Prentice Hall, New York, 1990.
3. J-M. Cordier and T. Porter. Shape Theory: Categorical Approximations Methods. Ellis Horwood Ltd, 1990.
4. D. P. Ellermann. Category Theory and Concrete Universals. ERKENNTNIS, 28. No.3,1988. p.409–429.
5. M. R. Garey and D. S. Johnson, “Computers and Intractability - A guide to the Theory of NP-Completeness”, Bell Laboratories Murray Hill, New Jersey, 1979.
6. L. A. S. Leal; P. B. Menezes; D. M. Claudio and L. V. Toscani. Categorias dos Problemas de Otimização. In: XIV SBES - Workshop on Formal Methods - WFM2000, 3. Oct. 04–06, João Pessoa, Paraíba. Proceedings... Brazil: SBC, 2000. p.104–109.
7. L. A. S. Leal; P. B. Menezes; D. M. Claudio and L. V. Toscani. Optimization Problems Categories. In: Formal Methods and Tools for Computer Science - EUROCAST’2001. Las Palmas de Gran Canaria, Canary Islands, Spain, R.Moreno-Díaz and A.Quesada-Arencibia Eds., 2001. p.93–96,
8. L. A. S. Leal; P. B. Menezes; D. M. Claudio and L. V. Toscani. Optimization Problems Categories. EUROCAST’01, LNCS 2178. Springer-Verlag, R.Moreno-Díaz, B. Buchberger and J-L. Freire Eds., 2001. p.285–299.
9. L. A. S. Leal. Uma fundamentação Teórica para a Complexidade Estrutural de Problemas de Otimização. (PhD Thesis), Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, 2002. 115p.
10. L. A. S. Leal; D. M. Claudio; P. B. Menezes and L. V. Toscani. Modelling the Approximation Hierarchy to Optimisation Problems Through Category Theory. International Journal of Computing Anticipatory Systems, v. 11, Belgium, 2002. p.336–349.
11. C. H. Papadimitriou, “Computational Complexity”, Addison-Wesley Publishing Company, 1994.
12. C. Rattray. Identification and Recognition Through Shape in Complex Systems. LNCS 1030, Springer-Verlag, 1995. p.19–29.
13. C. Rattray. Abstract Modelling Complex Systems. Advances in Computing Science, Systems: Theory and Practice. R. Albrecht, Ed., 1998. pp. 1–12.

A Formulation for Language Independent Prelogical Deductive Inference

Josep Miró

Departament de Ciències Matemàtiques i Informàtica
Universitat de les Illes Balears
07122 Palma de Mallorca, SPAIN
dmijmn0@uib.es

Abstract. A reality, supposedly described by an Object Attribute Table, and an unspecified language capable to describe subsets of objects are assumed. The requirements for an statement to be true are investigated, using set theoretical concepts only. The results lead to a sufficient condition for a statement to be deduced from a set of other statements (premises), by means of some algorithmic approach.

1 Introduction

The process of deductive inference has been very important for the scientific development of mankind. It is a central topic studied in logic. It is an extended belief that deductive inference is justified by logic. In fact, advances in the art of deductive inference have usually appeared in the academic field of logic, authored by specialists in this field. One of the characteristics of every logic is that logical expressions must be written in a specific language.

Recently another language has been developed to describe the realities supported by data banks in terms of discrete multivalued attributes [1]. The expressions are far from being binary. This fact has brought up the question of whether the processes of inference are possible when expressions are written in this fashion, and if so, what algorithms might accomplish them.

It is assumed here that the reality may be modeled by a set of objects, a set of attributes and an object attribute table. In this paper an approach to deductive inference is presented without a strong language specification. The main requirement is that the language should be able to describe every subset of objects. This formulation makes use of no logical rule, accepted a priori, and its approach is based on set theoretical concepts only. The present formulation leads to a sufficient condition, whose fulfillment assures the possibility of deductive inference.

1.1 Primary Knowledge

The main objective of inference processes is to issue new statements about a reality. This reality must be known, either through previous statements or through

direct acquaintance with it. Knowledge acquired directly from a reality is qualified here as *primary*.

The body of primary knowledge is made up of different sensations and other information whose structure may be considered as a data bank. This reality consists of a number of objects which exhibit a number of characteristics and relations among them. For the purpose of this paper we will assume that the initial reality is made up of a set of objects $O = \{o_1, o_2, \dots, o_s\}$ and a set of characteristics or attributes $R = \{r_1, r_2, \dots, r_n\}$. For example, O might be a set of persons in a room, and the attributes might be their considered features, for example *age, eyes color, height, weight, civil status etc.* Each characteristic $r_j, j = 1, 2, \dots, n$ may have a set of a n_j possible set of values, or *value range*, $W_j = \{w_{j1}, w_{j2}, \dots, w_{jn_j}\}$. The set of values of the attribute $r_j = \text{hair color}$ might be: $W_j = \{\text{black, brown, red, blond, gray, white}\}$, a set of six values. Therefore set R induces a set of sets, $W = \{W_1, W_2, \dots, W_n\}$. In this paper it is assumed that O, R and $W_j, j = 1, 2, \dots, n$, are finite sets. The set $\{R, W\}$ is called the *schema* of the data bank. For the purposes of this discussion it is considered that the data bank contains the values of each characteristic $r_j, j = 1, 2, \dots, n$, for every object $o_i, i = 1, 2, \dots, s$.

For convenience the data may be displayed in a table showing the values t_{ij} , that every attribute, r_j takes for every object o_i . These values, t_{ij} , may be coceived as a matrix, one row per object and one column per attribute. The values $t_{ij}, j = 1, 2, \dots, n$, of row o_i may be considered as the components of a vector, v_i . Each row of the table is displayed as a vector.

Given a schema every possible object description may be considered as one element of the set of possible vectors, U , given by $U = W_1 \times W_2 \times \dots \times W_n$, where $\times$ is the symbol for the cartesian product. The power set $P(U)$ represents the set of all possible table matrices for the given schema.

Since each row may be considered as a the vector describing one object, the objects described by equal vectors are indiscernible. Let an *indiscernibility* binary relation ρ be defined. $[o_i \rho o_k] = 1$ if $v_i = v_k$, and $[o_i \rho o_k] = 0$ if $v_i \neq v_k$. It can be proved that ρ is an equivalence relation and therefore ρ induces a classification on the set of objects O .

The quotient set $O/\rho = D = \{d_1, d_2, \dots, d_m\}$ is a set of m classes, each class being made up of all objects that cannot be discerned one from the other. For every class, $d_i, i = 1, 2, \dots, m$, there is one vector of attribute values, v_i , shared by all the objects in the class. The table may be rewritten, one line per class. For convenience d_i , which is actually a class, will be referred to simply as an *object*, since it may be considered to be an object representing the whole class. Borrowing algebraic nomenclature D is called the *domain* of the table. R is called the *range* or *codomain* of the table. Such a table, called *context* by Wille [2] and *information system* by Pavlak [3], will be referred to here as an *Object Attribute Table* or *OAT* in short. Thus, $\langle D, R, V_k \rangle$ specifies an OAT of domain D , range R and matrix formed by the set of vectors $V_k \subseteq U$ or $V_k \in P(U)$.

Actual realities are described by objects, attributes, and relations among objects. An OAT is a set of functions $D \longrightarrow \{0, 1\}$. A ν -ary relation may be

described by a function $D^\nu \rightarrow \{0,1\}$, and a set of ν -ary relations may be described by a similar table with a domain of ν -tuples. It is assumed here that the considered realities are described by tables of this sort. For convenience only the case of the *OAT*, for which $\nu = 1$, is considered in this paper, although the results may be extended to other values of ν .

1.2 Symbolic Representations

In the *OAT* defined by $\langle D, R, V_k \rangle$ all objects d_i are discernible. The power set $P(D)$ has 2^m elements. Every subset $D_i \subseteq D, i = 1, 2, \dots, 2^m$, is an element of $P(D)$. Therefore every *OAT* $\langle D_i, R, V_k \rangle$, is a *part* of the total $\langle D, R, V_k \rangle$.

Assume that every D_i has a symbol, or symbolic expression, σ_i representing it. The set of symbols $\Sigma = \{\sigma_i, i = 1, 2, \dots, 2^m\}$ makes up a symbolic system of representation of $P(D)$. $P(D)$ is a Boolean algebra, $\langle P(D), \cup, \cap, \sim, D, \emptyset_D \rangle$, where $\sim$ is the symbol for complementation, $\emptyset_D$ is the symbol for the empty set of D . The symbolic representation induces a Boolean algebra on Σ , $\langle \Sigma, +, \cdot, \sim, \sigma_D, \sigma_{\emptyset_D} \rangle$.

If the correspondence $D_i - \sigma_i$ is a random one, the operations on Σ must be described table wise. In some cases what is known about the symbol structure allows the following:

1. For every σ_i there is a complementary $\sim \sigma_i$ such that $\sigma_i + \sim \sigma_i = \sigma_D$, $\sigma_i \cdot \sim \sigma_i = \sigma_{\emptyset_D}$.
2. For all σ_i and σ_j it is possible to determine: $\sigma_i + \sigma_j$ and $\sigma_i \cdot \sigma_j$.
3. The result of operations $+$, $\cdot$ and $\sim$ may be determined by means of a computational process, without resorting to the operation tables.
4. The symbol not only represents an entity, but it conveys information about it in terms of its attributes.

In such a case, in this paper, the system of representation is called *descriptive* with respect to $+$, $\cdot$ and $\sim$, and σ_i is a *description* of D_i .

Since the purpose of a descriptive system is to describe a reality, or a part of it, not only representing it by a symbol but conveying knowledge about it, it follows that it must be able to describe a singleton $\{d_i\}$. As a consequence d_i must be described by a symbol or symbolic expression η_i composed in some way by the symbols of the values of the attributes of d_i . Since the structure of η_i is not predetermined, there is no reason why it should be unique; therefore nothing is said about it. The symbolic representation η_i is qualified as *elementary* because is valid for *only one* indiscernible object class d_i .

Since η_i describes d_i , η_j describes d_j and $+$ in Σ is the symbol of $\cup$ in $P(D)$ it follows that $d_i \cup d_j$ may be described by $\eta_i + \eta_j$. Therefore the symbol $+$ may be used in an expression describing a subset $D_h \in P(D)$. Similar arguments will lead to similar conclusions for $\cdot$ and $\sim$.

For every $D_h \in P(D)$,

$$D_h = \{d_i \dots d_j\} = \bigcup_i^j \{d_k\}$$

therefore D_h must be described at least by the *expanded* expression

$$\delta_h = \sum_i^j \eta_k$$

where δ_h is the simple symbol used here to denote an expanded expression.

However, there are no reasons to assume that δ_h is the only possible description of D . The binary relation between distinct expressions describing the same D_h , for all h , is again an equivalence relation. All the expressions describing the same D_h are *equivalent*. δ_h is the *expanded expression* of the class. In every class of equivalent expressions there is an expanded one.

Let E be the set of all expressions in a descriptive system, $e \in E$ denotes an expression in E . When it is desired to specify the entity being described by e , it will show up in parenthesis. So, $e(D_h)$ denotes an expression describing D_h . Subscripts might be used to identify individual expressions. The duality inherent in $\langle \Sigma, +, \cdot, \sim, \sigma_D, \sigma_{\emptyset_D} \rangle$ implies a duality in E , not considered here.

Let e_i and e_j be two arbitrary expressions of an descriptive system E , let $\preceq$ be the relation induced on E by the inclusion relation $\subseteq$ in $P(D)$, and $\delta_h = \Sigma_i^j \eta_k$. The following results are almost immediate:

1. $\eta_j \preceq \delta_h$.
2. $\preceq$ is a partial order, therefore, for all i, j , if $e_i \preceq e_j$, then $\eta_k \preceq e_i$ implies that $\eta_k \preceq e_j$.
3. $e_i \preceq e_j$ if and only if $e_i \cdot e_j = e_i$ and $e_i + e_j = e_j$.

1.3 Bases

The set of the attributes whose values appear in an expression, is called the *basis of the expression*. It has been considered so far that e_i is an expression using the symbols of values of all attributes $r_j, j = 1, 2, \dots, n$, of the schema. However, it may happen that e_i is equivalent to another expression e_k , that uses only a subset of attributes $R_b \subseteq R$. In this case it is said that R_b is also a basis. A basis in terms of which an expression may be found for every subset of D , is called here a *universal* basis for D . The problem of finding a suitable basis to describe a subset $D_i \in P(D)$ has been studied by many authors as Fiol [4], Michalski [5], Pavlak [6], Quinlan [7] and myself [8], and it falls out of the scope of this paper.

2 Universe

The set of all possible vectors, introduced above, is called a *universe* U . The following statements are immediate:

- 1.- Every set of attribute values, one value per attribute, that is, every vector v_i in an OAT, is an element of the universe U .
- 2.- Every basis $R_b \subseteq R$ of a schema induces a corresponding universe U .

Since a subset of the domain D_i is corresponded by a subset of vectors it follows, on the one side, that

3.- A subset $D_i \in P(D)$, $D_i = \cup_i^j \{d_k\}$, is described by $\sum_i^j \eta_k$ and induces a subuniverse, represented by $U(D_i)$.

On the other side every η_i describes a d_i , therefore it follows also that:

4.- η_i induces an *elementary subuniverse*, or singleton $U(\eta_i)$ consisting of an unique vector v_i .

Consequently :

5.- $\delta_h = \eta_i + \dots + \eta_j$ induces also a subuniverse, represented by $U(\delta_h)$.

6.- Every well formed expression e_j , equivalent to its expanded expression δ_j , induces a subuniverse, represented by $U(e_j)$.

Notice that $U(D_i)$ and $U(e_j)$ are not different kinds of subuniverses, but subuniverses of the same kind, although differently induced. Domain subsets and expressions are totally different concepts, but they may be related through the subuniverses they induce. This is stated formally in the sequel.

Theorem 1. *Every subset $D_i \in P(D)$ induces a subuniverse $U(D_i)$ and every expression $e_j \in E$ induces a subuniverse $U(e_j)$.*

The following results can be easily proved:

1.- $U(e_x + e_y) = U(e_x) \cup U(e_y)$

2.- $U(e_x \cdot e_y) = U(e_x) \cap U(e_y)$

3.- $U(\sim e_k) = \sim (U(e_k))$

Where $\sim U(e_k)$ is the complement of $U(e_k)$ with respect to U , that is

$$\begin{aligned} U(e_k) \cup \sim U(e_k) &= U \\ U(e_k) \text{ cap } \sim U(e_k) &= \emptyset_U \end{aligned}$$

2.1 Definitions

Since for all i , η_i describes d_i and every d_i is discernible, it follows that η_i describes d_i and only d_i , that is, η_i *defines* d_i and it can be written

$$d_i = \{d_k \in D \mid \eta_i\}.$$

That is, η_i is a *defining expression*. When it is stated that e_i is a defining expression, it is meant that there is a $D_i \subseteq D$ such that $U(D_i) = U(e_i)$. This is usually written:

$$d_i = \{d_k \in D \mid e_i\}.$$

That is:

$$\begin{aligned} \forall d_i \in D, \exists \eta_k \mid U(\eta_k) &\subseteq U(e_i) \\ \forall \eta_k \preceq e_i, \exists d_k \mid U(d_k) &\subseteq U(D) \end{aligned}$$

In the particular case in which $D_i = D$, $U(D)$ is a particular subuniverse determined by the OAT. A different OAT with the same schema might be described by a different defining expression $e(D)$ and would induce a different subuniverse $U(D)$. In this case

$$D = \{d_i \in D \mid e(D)\}.$$

On the one side, $e(D)$ is a defining expression for D . On the other side, from the OAT, the expanded expression version of the class of expressions equivalent to $e(D)$ may be easily derived. In other words, D is the subject about which $e(D)$ is predicated in a defining fashion.

Theorem 2. *For all i , let δ_i be a defining expression for D_i*

$$U(D_i) = U(\delta_i)$$

Proof. Since, for all j , η_j describes d_j and d_j is discernible, η_j describes d_j *only*, that is η_j does not describe d_h , $j \neq h$. Therefore η_j *defines* d_j and $U(\eta_j) = U(d_j)$. Since $\delta_i = \sum_k \eta_k$ and every element $d_j \in D_i$ is described by η_j , it follows that $U(D_i) = U(\delta_i)$.

In particular it is so for $D_i = D$

2.2 Truth and Verification

The following formal statements define the familiar concept of *truth* as a qualification of a sentence which describes some aspect of an objective reality. From the above discussion it follows:

- 1.- A domain element d_k verifies the expression e if and only if $U(d_k) \subseteq U(e)$.¹
- 2.- Expression e is verified by D_i if and only if for all $d_k \in D_i$, d_k verifies e .
- 3.- Expression e is a *true expression for D_i* , or a *truth of D_i* , if and only if e is verified by D_i .² This can be also written:

$$\forall d_k \in D_i, \eta_k \preceq e.$$

Therefore $U(\eta_j) \subseteq U(e)$.

4. Two expressions e_i and e_j are *consistent* if both may be simultaneously true of some reality.

From these concepts and the theorems on universes next theorem follows.

Theorem 3. *For all $d_k \in D_i$, expression e is true if and only if $U(D_i) \subseteq U(e)$.*

In particular for $D_i = D$, e , is true of D if and only if $U(D) \subseteq U(e)$. In this case:

- 1.- Expression e is true of D .
- 2.- A true expression e for D *describes* D , but it does not define it.
- 3.- Let δ be a defining expression for D , e is a true expression for D , if and only if $U(\delta) \subseteq U(e)$.

When a defining expression δ is known for D , condition 3 alone is enough to determine whether e is true of D . In every day language neither concept $U(D)$

¹ Colloquially “to verify” means “to check”. Here it has the meaning of “to corroborate”, “to validate” or “to make true”

² The use of both propositions is grammatically correct, “considering” or “corresponding to” are meanings of *for*; *of* has the sense of “inherent” or “relating to”. Both are pertinent.

nor $U(e)$ are used. Consequently, the possibilities of condition 3 are disregarded. Unfortunately, only expressions whose truth is affirmed or assumed are used without reference to their universes. This is why the truth of a statement is checked by trying to find a counter-example instead of using condition 3, which may be stated formally as follows:

Theorem 4. *Let $e(D_i)$ be a true expression for D_i , and δ_{D_i} a defining expression for D_i , then $U(\delta_{D_i}) \subseteq U(e_{D_i})$.*

3 Deduction

Deduction is an inference process whose objective is to solve the following problem: Given a finite set of expressions for D_i , $E = \{e(D_i)_1, e(D_i)_2 \dots\}$, called *premises*, whose truth and consistency is either affirmed or assumed, with respect to a given context (OAT), determine whether an expression e , called *conclusion*, not in E , is also true of D_i , in the same context, as a consequence of E .

The inferred conclusion is said to be a *derived* or a *deduced* expression.

Logic is an example of deductive processes. Logic makes use of logical rules. Each rule states that if two premises satisfy certain conditions, then an expression, called *logical conclusion*, follows. Such deduced conclusion may be incorporated to E , enlarging the set of premises, and the rules may be successively applied until the desired conclusion e_k is derived. The sequence of intermediate results is the *proof* of the conclusion. When no logical proof is found for e_k , it may be either because the expression is not true or because the proof has not been found yet, or because its truth is not a consequence of the premises. The number of logical rules assumed to be true without a proof is not nil. The rule called *modus ponendo ponens* is usually assumed without a proof, and the other rules may be logically derived later.

Prelogical is the qualification given here to an approach to deductive inference that doesn't require the use of a logical rule, nor the classical search for a proof. It is based on the following fundamental theorem.

Theorem 5. *Let $e_i, e_j \dots e_h$ be expressions assumed true of D_x in a certain context. A sufficient condition for e_k to be true of D_x in the same context is:*

$$U(e_i) \cap U(e_j) \cap \dots \cap U(e_h) \subseteq U(e_k)$$

Proof. Since $e_i, e_j, \dots e_h$ are true of D_x , by Theorem 4

$$U(D_x) \subseteq U(e_i) \quad U(D_x) \subseteq U(e_j) \quad \dots \quad U(D_x) \subseteq U(e_h)$$

therefore

$$U(D_x) \subseteq U(e_i) \cap U(e_j) \cap \dots \cap U(e_h)$$

and if

$$U(e_i) \cap U(e_j) \cap \dots \cap U(e_h) = U(e_\cap) \subseteq U(e_k),$$

then e_k is true of D_x .

3.1 Comments

The test to determine whether or not e_k is inferible from $e_i, e_j, \dots, e_h$ reduces to find out whether a set $A = U(e_{\cap})$ is a subset of $B = U(e_k)$. There are four well known ways to test whether $A \subseteq B$:

1.- If $A \subseteq B$ then $A \cup B = B$.

Sets A and B are described by their corresponding expressions. The test requires to compare the expression for $A \cup B$ and the expression for B ; being able to determine when the two expressions are equivalent. This is not a simple matter, as the author considered years ago [9].

2.- If $A \subseteq B$ then $A \cap B = A$.

The previous comment applies here too.

3.- If $A \subseteq B$ then $\sim A \cup B = U$.

This test requires to recognize when an expression induces the whole universe.

4.- If $A \subseteq B$ then $A \cup \sim B = \emptyset$.

This test requires to recognize when an expression induces the empty set of the universe. This is the test used in resolution techniques by using dual expressions in *clausal form*. It follows from here that logic is not required to justify the fundamentals of resolution algorithms.

4 Conclusions

From the discussions above it follows that deductive inference is possible if the two following conditions are satisfied:

1.- The symbolic expressions allow the description of every subset of the universe in such a manner that in the expression realm, E , operations $+$, $\cdot$, $\sim$ are established in such a way that for all e_x and e_y

$$U(e_x) \cup U(e_y) = U(e_x + e_y)$$

$$U(e_x) \cap U(e_y) = U(e_x \cdot e_y)$$

$$\sim U(e_x) = U(\sim e_x)$$

and this is so regardless of the expressions structure.

2.- It is possible to recognize either when either two expressions are equivalent, or when an expression describes the whole universe, or when an expression describes the empty set of the universe.

Acknowledgements. This work has been supported by the Universitat de les Illes Balears through the UIB 2003/11 project.

References

1. M. Miró-Juliá A contribution to the study of multivalued systems. Doctoral thesis. Universitat de les illes Balears, (2000)

2. Wille, R.: Restructuring Lattice Theory: an Approach based on Hierarchies of Concepts. *Ordered Sets*, Reidel Publishing Company (1982) 445–470.
3. Pawlak, Z.: Information System theoretical Foundation. *Information Systems*, **6** (1981), 205–218
4. Fiol, G. and Miró, J.; Theoretical Considerations about Subset Descriptions. *Lecture Notes on Computer Science* **763** (1993), 54–65
5. Michalski, R.S.: A Theory and Methodology of Inductive Learning. *Artificial Intelligence* **20** (1983) 111–161.
6. Pawlak, Z.: *Rough Sets: Theoretical Aspects of Reasoning About Data*. Kluwer Academic Publisher (1991).
7. Quinlan, J. R.: Induction of Decision Trees. *Machine Learning* **1** (1986) 81–106.
8. Miró Nicolau J.: On defining a Set by a Property (Technical report). Universitat de les Illes balears (1987)
9. Miró, J. and Miró-Julià, M.: Equality of Functions in CAST. *Lecture Notes in Computer Science* **1030** (1995) 129–136.

Multi-agent Simulation in Random Game Generator

Takuhei Shimogawa

School of Management, Faculty of Economics, Musashi University,
1-26-1 Toyotama-Ue, Nerima, Tokyo 176-8534, Japan
smgw@cc.musashi.ac.jp

1 Introduction

On the purpose to construct an agent-based simulation system derived from game theory, it seems to be quite significant to establish mathematical tools. It is also natural to consider them to be the contribution of applying method of systems theory to the subjects in social science.

The author proposed a formulation/formalization of required/supposed situation in : pp125-138 on [7]. This is the succeeding part of the study, based on concepts in the article.

This paper is devoted to the issue of : the study for some concrete observation of “agents” who can reason a solution of a given ‘game’ based on the ‘information structures’ they own. To be more concrete, the author provides several ideas of “spontaneous” assumption for the agents who are being into the games, and also, using the ideas, the manner of revising the knowledge about “others’ knowledge” which roles quite important part of this study.

2 Games and Models

In [7], a formal definition to describe a game and players each of which has some of *information* of the others’. The situation was named *a model of the game*. (Please refer to [7] for the detailed definitions.)

Definition 1 (Model). *Let us be given a game $\langle \Gamma, v, f \rangle$. Also, for each i in $\text{cod}(f)$ (codomain of f in the game, namely, the players in it), let him/her have his/her information structure $\langle \Omega, P_i \rangle$ respectively. A strategy profile of an agent $i \in \text{cod}(f)$ is a function $s_i : \Omega \rightarrow \Sigma_i$. A model of the game $\langle \Gamma, v, f \rangle$ is a tuple $\langle \Omega, \{P_i\}_{i \in \text{cod}(f)}, \mathbf{s} \rangle$, where $\mathbf{s}$ is said to be strategy profile of the model, which maps on Ω into $\prod_{i \in \text{cod}(f)} \Sigma_i$ satisfying $P^i(\mathbf{s}(\omega)) = s_i(\omega)$ where P^i is a projection.*

Based on this notion, the two “extreme” situation were clearly described, as the notions of *primitive model* and *perfect information model*.

Definition 2 (Primitive Model). *Let us be given a game $\langle \Gamma, v, f \rangle$. The primitive model for the game, denoted by $\langle \Omega^{\Gamma, f}, \{\dots, P_i^{\Gamma, f}, \dots\}, \mathbf{s} \rangle$. is a model such*

that $\Omega^{\Gamma, f}$ is primitive world and for all $i \in \text{cod}(f)$, $s_i : \Omega^{\Gamma, f} \rightarrow \Sigma_i$ is surjective and

$$\omega' \in P_i^{\Gamma, f}(\omega) \Leftrightarrow s_i(\omega) = s_i(\omega').$$

Definition 3 (Perfect Information Model). A model $\langle \Omega, \{P_i\}_{i \in \text{cod}(f)}, \mathbf{s} \rangle$ of a game $\langle \Gamma, v, f \rangle$ is said to be perfect information model provided $\Omega = \Omega^{\Gamma, f}$ and for every $i \in \text{cod}(f)$ $P_i(\omega) = \{\omega\}$ for all $\omega \in \Omega$.

It would be possible to say that, the two models just induce trivial behaviour of players, because : a) primitive model is a description of condition “there are only players who do not have any information of others” and b) perfect information model is for the condition “there are only players who own the ability to expect others’ behaviour perfectly”. The author tries to introduce some spontaneous dynamics between these extreme cases.

3 Knowledge about “Rationality” of Others

The main issue of this paper is to “provide” some natural way for *artificial* agents to reflect the knowledge about others’ throughout the environment of playing games. The knowledge about others’ knowledge has some essential role in a game, as is commonly mentioned in game theory. So, the question here is : how can they(the “artificial” agents) obtain/dismiss that kind of knowledge, or say, what kind of formulation is appropriate to describe the intuition ?

In the repeated playing of “various games” for the agents, they should be considered to be clever enough to utilize their experiences of the past for the next game. In the simulation of repeated games by fixed set of agents, each of them is supposed to gain some conviction concerning to others’ rationality, others’ conviction about others’ rationality, ... and so on. In order to implement that kind of “reality”, the following notions are to be used. Later on, we will see one demonstration of “learning about others’ knowledge” using one simple example.

Definition 4. Let the set of indices $\{1, 2, \dots, n\}$ of the agents be given. An information structure of the agent i is a pair $\langle \Omega, P_i \rangle$, where Ω is a set of the state of the world, and P_i denotes a function : $\Omega \rightarrow 2^\Omega$ which is called probability correspondence.

This is the generalized definition which is ordinarily used, but in this paper, let us consider the set of the state of the world to be as follows:

$$\Omega \triangleq \{\omega \mid \omega : V \rightarrow \{0, 1\}\}$$

Basically, $\omega \in \Omega$ represents a precise assignment of truth value to all of the basic statement of the world and V is to be the set of the statements. In this paper, it is supposed to represent a *strategy profile*. To be more precise, V essentially equals to $\text{Sub}(\Gamma) - G(\Gamma)$ (please refer to [7]), each of which element only two

action: $\{0, 1\}$ is assigned. The common interpretation for P_i will also be employed. (That is, at the real state $\omega \in \Omega$ the agent i can only recognize that one of the states in $P_i(\omega)$ is considered to be true. In other words, $\omega_1, \omega_2 \in P_i(\omega)$ if and only if the agent i cannot distinguish ω_1 from ω_2 .)

Definition 5. Let an information structure $\langle \Omega, P_i \rangle$ of an agent i be given. The function $\theta_i^\omega : V \rightarrow \{0, 1\}$ is said to be the scope of the agent i at the state $\omega \in \Omega$ provided:

$$(\forall \omega_1, \omega_2 \in \Omega)(\omega_1 \cdot \theta_i^\omega = \omega_2 \cdot \theta_i^\omega \Leftrightarrow \omega_1, \omega_2 \in P_i(\omega)),$$

where the dot (in $\omega_1 \cdot \theta_i^\omega$ for example) denotes ordinary index-wise conjunction operation:

$$(\forall j \in V)(\omega_1(j) \wedge \theta_i^\omega(j) = (\omega_1 \cdot \theta_i^\omega)(j)).$$

This definition describes the scope which an agent can see in a particular state. In concrete, for the scope θ_i^ω of the agent i who is in the state ω , $\theta_i^\omega(n) = 1$ if and only if the statement n can be asserted or refused by the agent i .

If $\theta_i^{\omega_1} = \theta_i^{\omega_2}$ holds in arbitrary pair of states $\omega_1, \omega_2 \in \Omega$, we shall use θ_i to denote the *scope of the agent i* . Such a condition is to be described as : *The agent i has his/her scope θ_i* ”.

Proposition 6. If an agent i has his/her scope, then :

- 1) $\omega \in P_i(\omega)$ for all $\omega \in \Omega$,
- 2) $\omega' \in P_i(\omega)$ implies $P_i(\omega) \subseteq P_i(\omega')$

Proof. 1) $\omega \cdot \theta_i^\omega = \omega \cdot \theta_i^\omega$ always holds, hence from the definition of θ_i^ω , $\omega \in P_i(\omega)$.

2) Pick up $\omega'' \in P_i(\omega)$ arbitrarily. Then $\omega' \cdot \theta_i^\omega = \omega'' \cdot \theta_i^\omega$ from the definition of the “scope”. This implies $\omega' \cdot \theta_i^{\omega'} = \omega'' \cdot \theta_i^{\omega'}$ because of the assumption that “ i has the scope”, whence using the definition of $\theta_i^{\omega'}$ again, $\omega'' \in P_i(\omega')$ is obtained. $\square$

From 1) and 2) in the proposition 6, it is easily derived that :

- 3) $\omega' \in P_i(\omega)$ implies $P_i(\omega') \subseteq P_i(\omega)$.

Next definition is quite general: we do not need to take any assumption for Ω .

Definition 7 (From the reference [3], e.t.c..). For an information structure: $\langle \Omega, P_i \rangle$, we say it is *partitional* if both of following conditions are satisfied:

- 1) $(\forall \omega, \omega' \in \Omega)(P_i(\omega) = P_i(\omega') \text{ or } P_i(\omega) \cap P_i(\omega') = \emptyset)$ and
- 2) $(\forall \omega \in \Omega)(\omega \in P_i(\omega))$.

Namely, this definition says that P_i constructs a certain partition on Ω . Next statement is straightforward result:

Fact 8 (From the reference [3], e.t.c..). An information structure $\langle \Omega, P_i \rangle$ is *partitional* if and only if it satisfies:

- 1) $\omega \in P_i(\omega)$ for all $\omega \in \Omega$ and
- 2) $\omega' \in P_i(\omega)$ implies $P_i(\omega) \subseteq P_i(\omega')$. $\square$

Consequently, for an agent, having a particular scope in the meaning described above is sufficient condition for his/her information structure to be partitional.

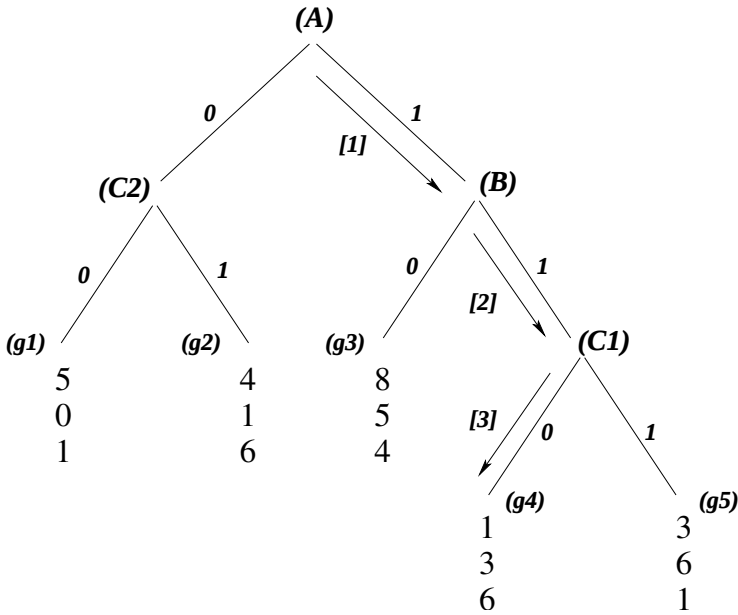


Fig. 1. An example

Definition 9. Let the information structure $\langle \Omega, P_i \rangle$ be given, the knowledge operator K_i derived by P_i is the function; $K_i : 2^\Omega \rightarrow 2^\Omega$ defined by the following manner:¹

$$K_i : \Omega \supseteq E \mapsto \{\omega \in \Omega \mid P_i(\omega) \subseteq E\} \subseteq \Omega.$$

A knowledge operator is uniquely determined by the given information structure $\langle \Omega, P_i \rangle$.

In the state ω , if $P_i(\omega)$ is contained in E , the agent i can be considered to know that E has occurred. Generally, the subset E of the state of the world Ω is called an event.

Using the notion of scope, a formal description of “process of repeated game” becomes suitable for the implementation of simulation systems. The usage will be introduced later.

4 Looking from an Example

Let us suppose first that the game displayed in Figure 1 is being played by the agents $\{A, B, C\}$ repeatedly. There are subgames : $\{(A), (B), (C1), (C2)\}$ each of which is owned by A, by B, by C and by C respectively. (That means, on

¹ For an information structure $\langle \Omega, P_i \rangle$, being partitional implies the following (famous) properties: (1) $K_i(\Omega) = \Omega$, (2) $K_i(E_1) \cap K_i(E_2) = K_i(E_1 \cap E_2)$, (3) $K_i(E) \subseteq E$, (4) $K_i(K_i(E)) = K_i(E)$ and (5) $\neg K_i(E) = K_i(\neg K_i(E))$, which represent the axiom schemata of modal system called S5.

a subgame, the owner of it can take an action.) Through this demonstration, the author tries to figure out some concrete idea concerning to what kind of “memory” to be obtained and being held by the agent in a simulation.

A game is basically the triple $\langle \Gamma, v, f \rangle$. (Precise definition is in [7].) Γ itself is the game tree in Fig.1 and for each subgame in it the index with “(” and “)” is attached. Actions are common: $\{0, 1\}$ for every agent, and edges of the game tree represents results of the actions. v denotes a sequence of payoff functions (v_A, v_B, v_C) : for instance, $v_C(g3) = 4$, $v_A(g5) = 3$ and so on.

Now *the states of the world* is to be introduced as being $\{(wxyz) \mid w, x, y, z \in \{0, 1\}\}$: it is thought of as the collection of *strategy profiles* (every player’s decision making). For instance, in Fig. 1, if the state ω is, say (1100), then it represents $s_A(\omega)((A)) = 1, s_B(\omega)((B)) = 1, s_C(\omega)((C1)) = 0$, and $s_C(\omega)((C2)) = 0$, where s_A and so on denotes a *strategy*.

4.1 First Stage

Suppose, the actual state is (1101). This means that A takes 1 on the subgame (A), B takes 1 on (B), C does 0 on (C1), 1 on (C2).

Before demonstrating the process, we might need to consider several assumptions. Firstly:

in a given games, if the depth of some subgame of the original game is 1, then the owner of the subgame is always able to be rational.

This seems to be mandatory, because otherwise the owner of the subgame (in this case, the player C), knowing what he/she is going to do, does not care about his/her payoff, which lead us to completely different context.

To describe this condition formally,

Assumption 1: Given a game $\langle \Gamma, v, f \rangle$, pick up arbitrary Γ'' from

$$D_1(\Gamma) \triangleq \{\Gamma' \in \text{Sub}(\Gamma) \mid (\forall \omega \in \Omega)(s_{f(\Gamma')}(\omega)(\Gamma') \in G(\Gamma))\}$$

For the subgame Γ'' , pick up $g \in \text{Sub}(\Gamma'')$, (every member of $\text{Sub}(\Gamma'')$ is a germ, from the assumption of Γ'') such that

$$(\forall g' \in \text{Sub}(\Gamma''))(v_{f(\Gamma'')}(g) \geq v_{f(\Gamma'')}(g'))$$

and make $E^{\Gamma''} \subseteq \Omega$ such that ;

$$\omega \in E^{\Gamma''} \Leftrightarrow s_{f(\Gamma'')}(\omega)(\Gamma'') = g$$

Finally,

$$\bigcup_{\Gamma'' \in D_1(\Gamma)} E^{\Gamma''} \tag{1}$$

In an actual state $\tilde{\omega}$ of given game, $\tilde{\omega} \in (1)$ is always assumed. The state (1101) of this case suffices this condition.

Also, suppose the game proceeds “step by step”, which intuitively means there is a time lag between actions, and the actions are taken from the top of

the game tree toward the bottom. The “process” is described as arrows with the number in square brackets attached on it in the Figure 1. Now, the actions of the game is being taken. Because of the assumption of state, the game goes like $[1] \rightarrow [2] \rightarrow [3]$ as is described in Fig. 1. The players have no knowledge about others in the first time play, so it is natural to consider that the *scope* of them is just restricted into the convictions of themselves: that is, $\theta_A = (1000)$, $\theta_B = (0100)$ and $\theta_C = (0011)$.

If we are concerned to some “real world”, it seems spontaneous to assume that there is some case in which each player can observe others’ action. Assuming this in this demonstration, the scope of each player is being revised in accordance with the process of the game, which is described as follows:

	θ_A	θ_B	θ_C
(Before A’s action)	(1000)	(0100)	(0011)
$[1]$	(1000)	(1100)	(1011)
$[2]$	(1100)	(1100)	(1111)
$[3]$	(1110)	(1110)	(1011)

Fig. 2. Revision of Scopes

Intuitively, each player’s probability correspondence gets more precise with the process goes. In this case, after the process $[2]$, C’s scope gets (1111) which means he has now understood everything about this game.

Fact 10. *In this game, after $[2]$ is done,*

$$(1101) \in K_B(E_{C1})$$

holds. Where E_{C1} represents the event(subset of the states of the world) of “sub-game (C1) occurs”.

Proof. It is sufficient to show $P_B((1101)) \subseteq E_{C1}$. In this point B’s scope is (1100), hence $P_B((1101)) = \{(1100), (1101), (1110), (1111)\}$. On the other hand, $E_{C1} = \{(11xy)\}_{x,y \in \{0,1\}}$. $\square$

So, after $[2]$, B is able to observe the action of C on (C1). Also:

Fact 11. *In this game, after $[2]$ is done,*

$$(1101) \in K_B K_C(E_{C1})$$

holds.

Proof. Let us check if $P_B((1101)) \subseteq K_C(E_{C1})$. $K_C(E_{C1}) = \{(1111)\} \cup \dots \cup \{(1100)\}$, because, in this point, $\theta_C = (1111)$ whence $P_C((1111)) = \{(1111)\}$ and so on (E_{C1} is already shown in previous fact). $\square$

This means that B *knows* that C knows that E_{C1} is the case in this point. Then, A and B observes C taking the action 0, resulting the germ ($g4$). The point is this: *B could observe the following points: after the process $[2]$, he/she knew the occurrence of (C1), and also, he/she knew that the occurrence was known by C.* Under the situation, B observed C taking 0. This seems to be sufficient for B to get convinced about C’s “rationality” in some particular circumstance. To be more concrete, B’s conviction after $[3]$ should be like “C can behave quite rationally when he/she is on a game of depth 1”.

4.2 Second and Third Stages

Now at least B can be thought to *know about C's rationality on a subgame of the depth 1*. This is the information which B is to remember throughout the iteration of playing games. Let us assume, that the next game is the same thing as was described in previous subsection.

Assumption 1 tells that the actual state needs to come from (1). In addition, there is another condition in this time. Player B sees the game in this time, finds the subgame C1 and decides not to take 1, because :

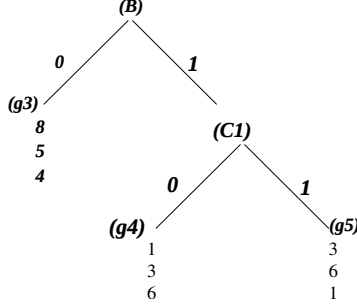


Fig. 3. Subgame B

it is irrational of B to take 1, knowing that C will take 0, resulting $v_B(g4) = 3$ which is worth than $v_B(g3) = 5$. Therefore B is now taking 0 on (B). On the other hand, player A sill remains uncertain that if B is “rational” or not, so it is still possible for A to take a chance: he/she might try to take action 0 on (A). Thus, let the actual state (0001) on this stage. We can then observe immediate result: (g2), causing no meaningful information as was occurred in the first stage.

The third stage: the actual state is, say, (1001). (A just tries his/her another action in this time.) The result is: (g3). In this stage, another “significant” thing happens: as we assumed before, player A can observe B behaving to take 0 in (B), resulting (g3) which is *backward induction solution* of subgame (B). Also, just like demonstrated in the fact 11, $(1001) \in K_A K_B(E_{(B)})$ holds after A’s action 1.

Here let us review the formalization of backward induction solution. ([7])

Definition 12 (Backward Induction Solution). *Given a game $\langle \Gamma, v, f \rangle$, the backward induction solution of the game, denoted by $BI(\Gamma)$ is the germ $g \in G(\Gamma)$ defined as :*

- 1) *If $\Gamma = \langle \rangle$ (namely, a germ) then $BI(\Gamma) = g$.*
- 2) *Otherwise, $BI(\Gamma) = BI(\Gamma')$, where*

$$\Gamma' \in Sub_1(\Gamma) \ \& \ (\forall \Gamma'' \in Sub_1(\Gamma)) [v_{f(\Gamma)}(BI(\Gamma')) > v_{f(\Gamma)}(BI(\Gamma''))].$$

Now in A’s mind, there should be a memory like “if some subgame which is ‘similar’ to (B) occurred, and also B knows that the subgame happened, he/she takes an action which will result backward induction solution of the subgame”.

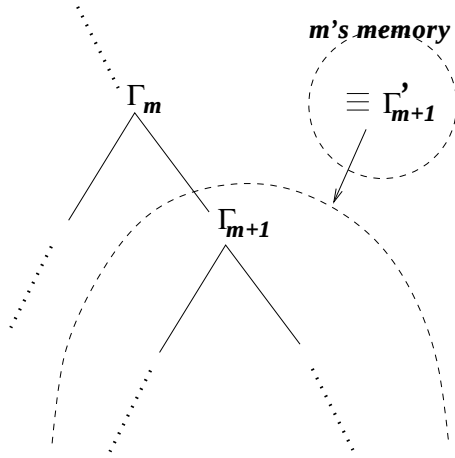


Fig. 4. If $\tilde{\omega} \in K_m K_{m+1}(E_{\Gamma_{m+1}})$ holds ...

4.3 Discussion

When considering ‘similarity’ mentioned in the demonstration of preceding subsections, the memory of player A to be held is : any subgame isomorphic² to (B) always results backward induction solution.

To be more general, if in the previous stage, a player m have seen the player $m + 1$ taking an action which caused at last a $BI(\Gamma'_{m+1})$, and in the present stage the player m sees Γ_m on which some action of himself results Γ_{m+1} which is isomorphic to Γ'_{m+1} , then his scope is to be constructed so as to estimate $BI(\Gamma_{m+1})$. Of course, if $\tilde{\omega}$ (the actual state) $\in K_m K_{m+1}(E_{\Gamma_{m+1}})$ is the case, the backward induction solution of Γ_{m+1} is to be considered actually to occur.

As is well-known, the assumption of “common knowledge” of the rationality of each agent just induces a trivial state. (In microeconomics, it is called *subgame perfect solution*.) As Aumann argued in [2], it usually doesn’t happen in the real world. The process demonstrated in preceding subsections is considered to be describing some reality, for, the “memory” of players and “dynamics” based on the memory is spontaneously connected. It is possible to claim this process(or aspect) to be implemented as a part of simulation systems on the purpose of this study. However in order to implement the system, the author needs to formalize them more precisely.

The author introduced “the dynamics of the state” in [7], where the process of a game and the “recognition” of the process is thought to be a spontaneous cause of it. This assumption is essentially thought of as the same as *substantive rationality* which is mentioned in [4] and [5]. Next stage of this study will include the observation how the assumption of substantive rationality affects the process described in subsection 4.1 and 4.2.

² Here the author considers isomorphic relation between trees : $\Gamma \equiv \Gamma'$, to be the existence of ‘isomorphism’ as that of normal sense.

5 Perspective

The method of “process/agent based simulation” is usually used from the standpoint of, for example, economics/social science. (Most significantly, Prof. Axelrod’s investigation ([8],[9] and so on.) The purpose is clearly the “observation” of the result. In contrast, in the systems theory, it seems that the “computational aspect” of the simulations (the author would like to call them “multi-interaction computation”) deserves being argued. This kind of view leads us to the significant conception of “adaptive nonlinear networks” provided by Prof. John Holland([6] for instance). The author thinks the standpoint employed in this study obviously differs from social science which basically just observes the result of simulation.

Game theory itself is, through this study, merely one guideline to depict the behavioral side of the society. The complexity of the society seems to depend not on “the behaviour” of agents itself. It seems to depend on the fact that the agents have “will” to be “better”, and behave based on some kind of “reflection” of the world inside themselves. The behavioral aspects are to be described in a “game”, and its “reflection of the world” is to be manipulated through “information structure”, “process logic” and so on. The systems theory employs them as “language” and manipulable notions.

References

1. Aumann, R.J.: Rationality and Bounded Rationality. Nancy Schwartz Lecture, Kellogg Foundation (1996)
2. Aumann, R.J.: Backwards Induction and Common Knowledge of Rationality. Games and Economic Behavior, Vol. 8. (1997), 6–19
3. Rubinstein, A.: *Modeling Bounded Rationality*. Zeuthen lecture book series. The MIT Press, Cambridge,Massachusetts,London,England, (1998).
4. Halpern, J.Y.: Substantive Rationality and Backward Induction. Economics Working Paper Archive at WUSTL, (1998) (<http://econpapers.hhs.se/paper/wpawuwpga/0004008.htm>)
5. Stalnaker, R.C.: Knowledge, belief and counterfactual reasoning in games. Economics and Philosophy, Vol. 12. (1996), 133–163
6. Arthur, W.Brian, N.Durlauf, Steven and A.Lane, David eds. : The Economy as an Evolving Complex System II . Addison Wesley(1997)
7. Moreno-Díaz, R., Buchberger, B. and Freire, J.L. eds. : LNCS2178 Computer Aided Systems Theory – EUROCAST2001. Springer(2001)
8. Axelrod, R.: Artificial Intelligence and the iterated prisoner’s dilemma. Discussion Paper 120, Institute of Public Policy Studies. University of Michigan, Ann Arbor, Mich.(1978)
9. Axelrod, R.: The Complexity of Cooperation, Agent-based Models of Competition and Collaboration, Princeton Studies in Complexity. Princeton University Press, New Jersey (1997)

The Zero Array: A Twilight Zone

Margaret Miró-Julià

Departament de Ciències Matemàtiques i Informàtica
Universitat de les Illes Balears
07122 Palma de Mallorca, SPAIN
`margaret.miro@uib.es`

Abstract. Descriptive knowledge about a multivalued data table or Object Attribute Table can be expressed by means of a binary Boolean based language. Efforts to design computer programs that determine declarations have been made. This endeavor is mainly directed to binary declarations.

This paper firstly proposes a multivalued language, based on arrays, that allows a multivalued description of the knowledge contained in a data table, by means of array expressions.

The zero array is singled out and its interpretation analyzed, following the discussion the projection arrays or project-ars are introduced.

Finally, a relation between formal concepts and the project-ars is examined.

1 Introduction

Descriptive knowledge about a multivalued data table, or Object Attribute Table (OAT), can be expressed in declarative form by means of a binary Boolean based language. Computer programs that determine declarations have been designed. Directly or indirectly, work by Michalski [1], Quinlan [2], Pawlak [3], Skowron [4], Miró [5], Wille [6] and Fiol [7] has to do with this problem. However, their efforts are mainly directed to binary descriptions. Given a set of objects $D = \{d_1, d_2, \dots, d_m\}$ and a set of attributes $R = \{r_1, r_2, \dots, r_n\}$ a binary Object Attribute Table (OAT) can describe a situation. Subsets of D are described in terms of the attributes and their binary values. A subset of D may be described by a function $f : \{0, 1\}^n \rightarrow \{0, 1\}$.

Declarative expressions for a function of a binary OAT can be obtained by means of a Boolean algebra. A Boolean algebra is a convenient tool when considering attributes that take binary values. Can an equivalent algebra that handles non binary attributes be obtained?

Definition 1. Let $D = \{d_1, d_2, \dots, d_i, \dots, d_m\}$ be an ordered set, called domain, of elements d_i representing the m objects, let $R = \{r_g, \dots, r_c, \dots, r_a\}$ be a set of the g attributes or properties of the objects. The set of values of attribute c is represented by $C = \{[c_{n_c}], \dots, [c_j], \dots, [c_1]\}$. The elements of set C , $[c_j]$, are called 1-spec-sets since the elements are defined by means of one specification. An Object Attribute Table (OAT) is a table whose rows represent the objects,

Table 1. Object Attribute Table

	r_g	$\dots$	r_c	$\dots$	r_a
d_1	$[g_1]$	$\dots$	$[c_1]$	$\dots$	$[a_1]$
d_2	$[g_2]$	$\dots$	$[c_2]$	$\dots$	$[a_2]$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
d_i	$[g_i]$	$\dots$	$[c_i]$	$\dots$	$[a_i]$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
d_m	$[g_m]$	$\dots$	$[c_m]$	$\dots$	$[a_m]$

and whose columns represent the attributes of these objects. Each element $[c_i]$ represents the value of attribute r_c that corresponds to object d_i .

2 Multivalued Algebra

In order to handle the multivalued OAT, where attributes take values from a given set, a new multivalued algebra is proposed. The starting point is a multivalued language that describes all subsets.

2.1 Multivalued Language

It is well known that the set of all subsets of a given set C (the power set of C), $\rho(C)$, constitutes a Boolean algebra $\langle \rho(C), \cup, \cap, \hat{}, \emptyset, C \rangle$. If a symbolic representation or a description of subsets is considered, there is a parallel Boolean algebra $\langle \mathcal{S}_c, +, \cdot, \hat{}, \vee_c, \wedge_c \rangle$ defined on the set $\mathcal{S}_c$ of all possible symbols representing subsets of C . The zero of this algebra is $\vee_c$ (the symbol representing the empty set). The identity is $\wedge_c$ (the symbol representing set C).

Throughout this paper, the symbol $\rightsquigarrow$ may be read as: “is described by”. Therefore, $C_h \rightsquigarrow c_h$ expresses: “subset C_h is described by symbol c_h ”. The symbolic representations of regular set operations complement ($\hat{}$), union ($\cup$) and intersection ($\cap$) are:

$$\widehat{C_h} \rightsquigarrow \hat{c}_h \quad C_h \cup C_k \rightsquigarrow c_h + c_k \quad C_h \cap C_k \rightsquigarrow c_h \cdot c_k$$

This generic symbolic representation and its operations have been carefully studied in [8]. The octal code used throughout this paper is (S represents something):

0	is the symbol for	$\bullet, \bullet, \bullet$	4	is the symbol for	$S, \bullet, \bullet$
1	is the symbol for	$\bullet, \bullet, S$	5	is the symbol for	$S, \bullet, S$
2	is the symbol for	$\bullet, S, \bullet$	6	is the symbol for	$S, S, \bullet$
3	is the symbol for	$\bullet, S, S$	7	is the symbol for	S, S, S

Example 1. Let’s assume that C is the set of objects on my desk.

Step 1: Set C is described as an ordered string:

$$C = \{\text{book}, \text{keys}, \text{stapler}, \text{pencil}, \text{cup}, \text{tape}, \text{phone}\}$$

Step 2: Set C is split in segments of 3 objects

$$C = \{\bullet, \bullet, b, \mid k, s, p, \mid c, t, ph\}$$

Any subset of C will be represented by means of three symbols.

Step 3: In order to describe the subset formed by the objects: tape, book, keys, phone and pencil, C_h is written as:

$$C_h = \{\bullet, \bullet, b, \mid k, \bullet, p, \mid \bullet, t, ph\}$$

Step 4: The elementary symbols of all segments are represented following the octal code. Therefore the sequence 1 5 3, together with the stored ordered description of set C is a unique description of the subset C_h .

$$C_h \rightsquigarrow c_h = 1\ 5\ 3$$

2.2 Fundamental Concepts

All the concepts and operations introduced above make reference to only one set, that is, only one attribute. A multivalued OAT has more than one attribute.

Let $R = \{r_c, r_b, r_a\}$ be a set of 3 attributes whose attribute values are $C = \{[c_{n_c}], \dots, [c_2], [c_1]\}$, $B = \{[b_{n_b}], \dots, [b_2], [b_1]\}$ and $A = \{[a_{n_a}], \dots, [a_2], [a_1]\}$. The elements of sets C, B, A are 1-spec-sets (one specification). A 3-spec-set, $[c_k, b_j, a_i]$, is a chain ordered description of 3 specifications, one from set C , one from set B and one from set A .

The 3-spec-sets brings to mind the ordered triples that form the cartesian product. But the cartesian product has an undesirable property:

$$A \times B \neq B \times A.$$

Given an OAT with two attributes, *color of the eyes* and *color of hair*, whether a person has “blue eyes and blond hair” or “blond hair and blue eyes” is irrelevant. What is important are the values of the attributes: *blue*, *blond* and the order in which there are written: *eye color*, *hair color*.

Therefore, to avoid this unwanted characteristic, each spec-set represents itself and all possible permutations. Hence,

$$[c_k, b_j, a_i] = [c_k, a_i, b_j] = [b_j, c_k, a_i] = [b_j, a_i, c_k] = [a_i, c_k, b_j] = [a_i, b_j, c_k]$$

In all definitions that follow, $R = \{r_g, \dots, r_b, r_a\}$ is the set of g attributes whose attribute values are given by non-empty sets $G, \dots, B, A$ respectively.

Definition 2. The cross product $G \otimes \dots \otimes B \otimes A$ is the set of all possible g -spec-sets formed by one element of G , ..., one element of B and one element of A .

$$G \otimes \dots \otimes B \otimes A = \{[g_x, \dots, b_j, a_i] \mid [g_x] \in G, \dots, [b_j] \in B, [a_i] \in A\}$$

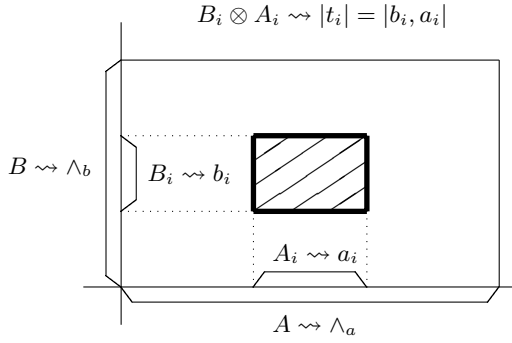


Fig. 1. 2-dimensional arrays

Once again, it is important to mention that the cross product is not the cartesian product. A g-spec-set represents itself and all possible permutations whereas the elements of the cartesian product are different if the order in which there are written varies. There is a need to determine an order in a g-spec-set. The basis T is an ordered chain $\langle G, \dots, B, A \rangle \equiv T$ which establishes the sequential order in which the spec-sets are always written. The basis considered throughout this paper is $T = \langle G, \dots, B, A \rangle$.

The set of all possible g-spec-sets induced by sets $G, \dots, B, A$ is called the **universe** and every subset of the universe is called a **subuniverse**.

Definition 3. Let $G_i \subseteq G, \dots, B_i \subseteq B, A_i \subseteq A$, an array $|t_i| = |g_i, \dots, b_i, a_i|$ is the symbolic representation of the cross product $G_i \otimes \dots \otimes B_i \otimes A_i$ where $G_i \rightsquigarrow g_i, \dots, B_i \rightsquigarrow b_i$, and $A_i \rightsquigarrow a_i$.

$$G_i \otimes \dots \otimes B_i \otimes A_i = \{[g_x, \dots, b_j, a_i] \mid [g_x] \in G_i, \dots, [b_j] \in B_i, [a_i] \in A_i\}$$

$$G_i \otimes \dots \otimes B_i \otimes A_i \rightsquigarrow |t_i| = |g_i, \dots, b_i, a_i|$$

An array is a symbolic representation of a special type of subuniverse called **cartesian** subuniverse which was proposed in [9].

Arrays are symbolic representations of cartesian subuniverses, 2-dimensional (two attributes) arrays can be represented graphically as shown in Fig. 1. To simplify the representation, the elements of subsets A_i and B_j are drawn together in a continuous subset, which needs not be the case.

Since the arrays describe cartesian subuniverses (subsets of spec-sets), regular set operations may be performed with them.

Let $|t_i| = |g_i, \dots, b_i, a_i|$ and $|t_j| = |g_j, \dots, b_j, a_j|$ be two arrays, the following operations between arrays are introduced:

- $\sim$ complement (symbolic representation of the complement of a subuniverse respect to the universe):

$$\sim (G_i \otimes \dots \otimes B_i \otimes A_i) \rightsquigarrow \sim |t_i|$$

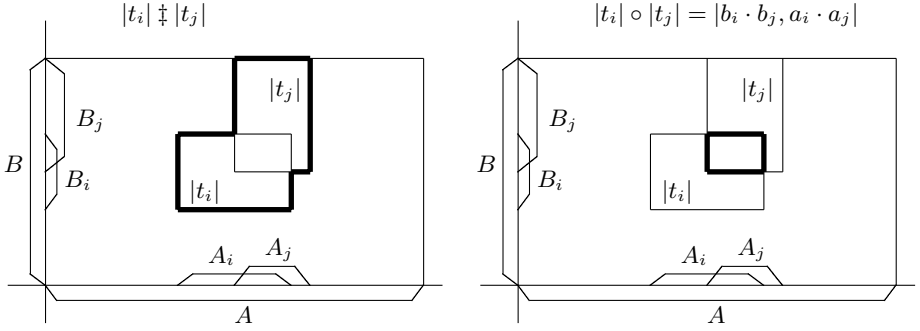


Fig. 2. 2-dimensional $\ddagger$ sum and $\circ$ product of arrays

- $\ddagger$ sum (symbolic representation of the union of two subuniverses):

$$(G_i \otimes \cdots \otimes B_i \otimes A_i) \cup (G_j \otimes \cdots \otimes B_j \otimes A_j) \rightsquigarrow |t_i| \ddagger |t_j|$$

- $\circ$ product (symbolic representation of the intersection of two subuniverses):

$$(G_i \otimes \cdots \otimes B_i \otimes A_i) \cap (G_j \otimes \cdots \otimes B_j \otimes A_j) \rightsquigarrow |t_i| \circ |t_j|$$

$$|t_i| \circ |t_j| = |g_i, \dots, b_i, a_i| \circ |g_j, \dots, b_j, a_j| = |g_i \cdot g_j, \dots, b_i \cdot b_j, a_i \cdot a_j|$$

The $\circ$ product of two arrays is an array.

All the results obtained by use of operations $\sim$, $\ddagger$ and $\circ$ on arrays are symbolic representations of subuniverses. If only two attributes are considered, these operations can be represented graphically as shown in Fig. 2.

Definition 4. An array $|t_y^E|$ is called an elementary array if it describes a subuniverse formed by only one g -spec-set.

$$G_y \otimes \cdots \otimes B_y \otimes A_y = \{[g_y]\} \otimes \cdots \otimes \{[b_y]\} \otimes \{[a_y]\} \rightsquigarrow |t_y^E| = |g_y, \dots, b_y, a_y|$$

where $G_y = \{[g_y]\} \rightsquigarrow g_y, \dots, B_y = \{[b_y]\} \rightsquigarrow b_y$, and $A_y = \{[a_y]\} \rightsquigarrow a_y$ (all subsets are 1-element subsets).

3 Expressions

Subuniverses can be described by algebraic expressions of arrays. An expression is a symbolic representation of a subuniverse. In general, an expression represents a partial reality included in an OAT.

Definition 5. Every combination of arrays using operations $\sim$, $\ddagger$ and $\circ$ (well formed formula) is called an expression E_i .

$$E_i = \sim |t_i| \ddagger |t_j| \circ |t_k| \dots$$

Generally, a subuniverse can be described by more than one expression. Distinct expressions that describe the same subuniverse are said to be equivalent (declaratively). The study of the equivalency between distinct expressions was originally proposed in [10] and explored in [8].

Definition 6. An expression E_i is called an array expression if it is written as a $\ddagger$ sum of arrays: $E_i = |t_z| \ddagger \cdots \ddagger |t_y| \ddagger \cdots \ddagger |t_x|$.

Expressions represent subuniverses, therefore an order relation equivalent to set inclusion may be introduced: $U_i \subseteq U_j \rightsquigarrow E_i \preceq E_j$. This order relation has been studied in [8] and has been used to find simplified equivalent expressions.

Definition 7. Let $E_i = |t_z| \ddagger \cdots \ddagger |t_y| \ddagger \cdots \ddagger |t_x|$, an array $|t_y|$ of the expression is a prime-ar (prime array) of E_i if there is no other array $|t_j|$ such that:

$$|t_y| \preceq |t_j| \preceq E_i$$

A prime-ar is a “largest” array contained in E_i .

The prime-ars will be represented by $|t_y^P|$. All of the prime-ars of an expression can be obtained by application of the prime-ar algorithm, given in [11].

An array expression is called elementary if each of the arrays in the $\ddagger$ sum is an elementary array: $E_i^E = |t_z^E| \ddagger \cdots \ddagger |t_y^E| \ddagger \cdots \ddagger |t_x^E|$. An elementary array is a minimal array in the sense that it cannot be expressed as a $\ddagger$ sum of other arrays. Therefore an elementary array expression is an expression made up of minimal arrays.

An array expression is called a prime-ar expression if each of the arrays in the $\ddagger$ sum is a prime-ar: $E_i^P = |t_z^P| \ddagger \cdots \ddagger |t_y^P| \ddagger \cdots \ddagger |t_x^P|$. A prime-ar is a maximal array in the sense that it is a largest array contained in the expression. Therefore a prime-ar expression is an expression made up of maximal arrays.

The $\ddagger$ sum of all the prime-ars of an expression E_i is called the all-prime-ar expression of E_i . The all-prime-ar expression is a unique expression, but the number of prime-ars may not be minimal.

In the following example, expressions for a given OAT are obtained.

Example 2. The following table, taken from [12] represents the general grades on conduct (c), diligence (d), attentiveness (a) and orderliness (o) given to a class at the Ludwig-Georgs-Gymnasium.

The students of the class constitutes the set $D = \{A, B, C, \dots, X, Z\}$ and the attributes are $R = \{c, d, a, o\}$, these attributes take values from A (best grade) to D (worst grade).

The octal code used for attributes c and a , that take 3 values is:

$$\bullet\bullet A \rightsquigarrow 1, \quad \bullet B \bullet \rightsquigarrow 2, \quad C \bullet\bullet \rightsquigarrow 4$$

The octal code used for attributes d and o , that take 4 values is:

$$\bullet\bullet\bullet A \rightsquigarrow 0\ 1, \quad \bullet\bullet B \bullet \rightsquigarrow 0\ 2, \quad \bullet C \bullet\bullet \rightsquigarrow 0\ 4 \quad D \bullet\bullet\bullet \rightsquigarrow 1\ 0$$

Table 2. Grades at the Ludwig-Georgs-Gymnasium

	c	d	a	o		c	d	a	o		c	d	a	o
Anna	C	D	C	D	Jurgen	B	B	C	B	Stefan	A	A	A	A
Berend	C	D	C	D	Karl	B	C	B	B	Till	A	A	A	A
Christa	B	B	B	B	Linda	B	A	B	B	Uta	A	A	B	B
Dieter	A	A	A	A	Manfred	B	B	B	B	Volker	B	B	C	B
Ernst	B	B	B	B	Norbert	C	C	B	B	Walter	C	D	C	D
Fritz	B	A	B	B	Olga	A	A	B	B	Xaver	A	A	A	A
Gerda	B	B	B	C	Paul	A	A	A	A	Zora	B	B	B	B
Horst	B	B	B	C	Quax	B	B	B	B					
Ingolf	B	C	C	B	Rudolf	C	D	C	C					

The elementary array expression for the table is:

$$\begin{aligned}
 E = & |1, 01, 1, 01| \ddagger |4, 04, 2, 04| \ddagger |4, 10, 4, 10| \ddagger |4, 10, 4, 04| \ddagger \\
 & \ddagger |2, 02, 2, 02| \ddagger |2, 02, 2, 04| \ddagger |1, 01, 2, 02| \ddagger |2, 01, 2, 02| \ddagger \\
 & \ddagger |2, 02, 4, 02| \ddagger |2, 04, 2, 02| \ddagger |2, 04, 4, 02|
 \end{aligned}$$

The all-prime-ar expression for the table is:

$$\begin{aligned}
 E = & |1, 01, 1, 01| \ddagger |4, 04, 2, 04| \ddagger |4, 10, 4, 14| \ddagger |2, 02, 2, 06| \ddagger \\
 & \ddagger |3, 01, 2, 02| \ddagger |2, 07, 2, 02| \ddagger |2, 06, 6, 02|
 \end{aligned}$$

The array theory allows the description of all knowledge contained in the table by means of a $\ddagger$ sum of arrays.

4 The Zero Array

There are two arrays that deserve special consideration. First, the identity array $\bigwedge$ that describes the universe:

$$U \rightsquigarrow \bigwedge = |\wedge_g, \dots, \wedge_b, \wedge_a|$$

Secondly, the zero array $\bigvee$ that describes the empty subuniverse:

$$\emptyset \rightsquigarrow \bigvee = |\vee_g, \dots, \vee_b, \vee_a|$$

In the developement of the array theory the following theorem was proven in [8]:

Theorem 1. *An array with a $\vee$ component is equal to $\bigvee$*

$$\forall b \quad |g_i, \dots, \vee_b, a_i| = \bigvee$$

where:

$$G_i \otimes \dots \otimes \emptyset_B \otimes A_i \rightsquigarrow |g_i, \dots, \vee_b, a_i|$$

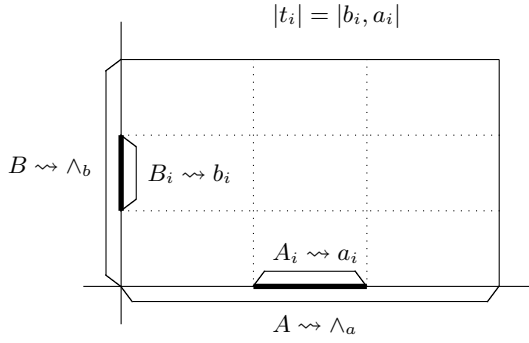


Fig. 3. 2-dimensional project-ars

Recall that $\vee_b$ is the zero of the Boolean algebra defined on the set $\mathcal{S}_b$ of symbols describing subsets of B .

This theorem gives rise to some interesting questions. Even though the cross product is not the cartesian product it inherits an undesirable property: the cartesian product of a set by the empty set is the empty set. If an OAT is considered, just because there is a missing piece of information can we say that we have no information at all?

Furthermore, if arrays with one $\vee$ component are equal to $\vee$ then:

$$|g_i, \dots, b_i, \vee_a| = |g_i, \dots, \vee_b, a_i| = |\vee_g, \dots, b_i, a_i| = \vee$$

In two dimensions an array $|b_i, a_i|$ can be graphically represented by a rectangle $b_i \times a_i$, as shown in Fig. 1. When one of the sides becomes zero, then the rectangle has zero area. In this sense $|b_i, \vee_a| = |\vee_b, a_i|$. But even though one of the sides is zero, the other is not. The array $|b_i, \vee_a|$ becomes a line of size b_i , whereas the array $|\vee_b, a_i|$ becomes a line of size a_i . Therefore there is a difference. These lines are the array projections or “project-ars”.

Definition 8. Given a cartesian array $|t_i| = |g_i, \dots, b_i, a_i|$, a first order project-ar, $|P^1|$, is an array with one $\vee$ component and $(g - 1)$ non-zero components, a second order project-ar, $|P^2|$, is an array with two $\vee$ components and $(g - 2)$ non-zero components, a n th order project-ar ($n < g$), $|P^n|$, is an array with n $\vee$ components and $(g - n)$ non-zero components.

The number of different project-ars of order n can be easily found by counting the number of ways n $\vee$ components can be placed in an array $|t_i| = |g_i, \dots, b_i, a_i|$. Depending on the location of the $\vee$ components, there are $\frac{g!}{n!(g-n)!}$ n th order project-ars.

The first order project-ars of array $|t_i| = |b_i, a_i|$ are: $|P_a^1| = |b_i, \vee_a|$ and $|P_b^1| = |\vee_b, a_i|$, shown on Fig. 3.

The starting point of the array theory is an OAT. Nothing has been specified about this table, but it is understood that it always is a complete table, that is, all objects have non zero values for all attributes. How can project-ars be obtained?

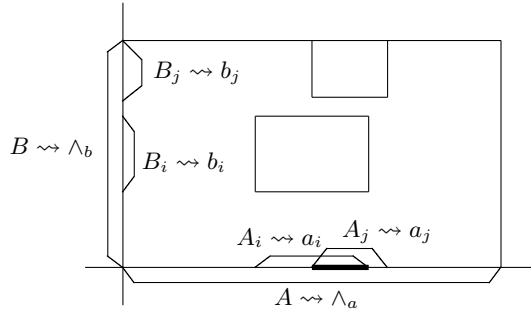


Fig. 4. A project-ar as a $\circ$ product of arrays

A simple way to obtain a project-ar is by means of the $\circ$ product of arrays. Fig. 4 enlightens this idea.

$$|b_i, a_i| \circ |b_j, a_j| = |b_i \cdot b_j, a_i \cdot a_j| = |\vee_b, a_i \cdot a_j|$$

The zero array $\vee = |\vee_g, \dots, \vee_b, \vee_a|$ is a unique array that describes the empty universe associated to g sets. The project-ars are arrays with $\vee$ components, in some way the project-ars can be considered as zero arrays of different degrees depending on the number of $\vee$ components. A n th order project-ar would correspond to a zero array of degree n . From another point of view, the project-ars are descriptions of a reality with missing attributes. The project-ars describe partial OAT's (incomplete tables).

5 Arrays and Formal Concepts

By application of the algebraic methodology described above, true declarative expressions of an OAT (context) may be obtained and may also be transformed into other equivalent expressions. The use of the theoretical results mentioned in this paper allows the treatment of the formal concept problem of a binary context, and extends it to multivalued OAT's, (multivalued contexts). To better compare project-ars and formal concepts, a brief outline of the main ideas is presented below.

5.1 Main Ideas

Given an OAT of domain D and range R , a formal concept is defined as the pair (D_i, R_i) , where $D_i \subseteq D$ (called extension) is a set of objects that exhibit a set of attribute values $R_i \subseteq R$ (called intension) and all these values only apply to those objects [12]. Set R is formed by the binary attributes, if attributes are multivalued then R is formed by all attribute values of all attributes. The

intension R_i is a subset of R , but nothing is said about the nature of the elements of R_i .

If the least and greatest concepts are not considered, then subset R_i is formed by none or one of the possible values of each attribute. At the most, R_i will have as many elements as attributes; at the least, R_i will have one element.

From the array point of view, every subuniverse U_i (every OAT) can be represented by an elementary array expression.

$$U_i \rightsquigarrow E_i^E = |t_z^E| \ddagger \cdots \ddagger |t_y^E| \ddagger \cdots \ddagger |t_x^E|$$

The elementary array $|t_y^E|$ describes a subuniverse, formed by only one g-spec-set (recall Definition 4).

$$\{[g_y, \dots, b_y, a_y]\} \rightsquigarrow |t_y^E| = |g_y, \dots, b_y, a_y|$$

The elementary array $|t_y^E|$ represents a set of attribute values, one and only one value per attribute is allowed. These attribute values apply to a subset of objects $D_y \subseteq D$, that is, D_y is the set of all objects whose description is given by elementary array $|t_y^E|$.

Therefore, given an OAT, an elementary array of the OAT together with the set of objects whose description is given by the elementary array form an array concept $(D_y, |t_y^E| = |g_y, \dots, b_y, a_y|)$. This array concept describes the formal concept $(D_y, R_y = \{g_y, \dots, b_y, a_y\})$. The same symbols $g_y, \dots, b_y, a_y$ are used to represent both the attribute values (in the formal concept) and the symbols representing these subsets (array theory).

There is some connexion between $R_y = \{g_y, \dots, b_y, a_y\}$ and the elementary array $|t_y^E| = |g_y, \dots, b_y, a_y|$. In some way, the symbolic array notation can be used to represent subsets of $R = G \cup \dots \cup B \cup A$. Since symbols are used to represent all the subsets of sets of the basis, the order fixed in array representation must be preserved, an empty space $\_$ is used to represent those attributes with no values. But the symbol that indicates no attribute values is $\vee$, therefore:

$$\begin{aligned} R_x &= \{g_x, \dots, d_x, c_x, a_x\} \rightsquigarrow |g_x, \dots, d_x, c_x, \_, a_x| = |g_x, \dots, d_x, c_x, \vee_b, a_x| \\ R_z &= \{g_z, \dots, d_z, b_z\} \rightsquigarrow |g_z, \dots, d_z, \_, b_z, \_ | = |g_z, \dots, d_z, \vee_c, b_z, \vee_a| \end{aligned}$$

Arrays and project-ars can describe the intensional part of a formal concept.

5.2 The Lattice Structure

All of the formal concepts of a context form a complete lattice. This lattice structure can be constructed by introducing a partial order relation $\leq$. Given two concepts (D_i, R_i) and (D_j, R_j) , $(D_i, R_i) \leq (D_j, R_j)$ if and only if $D_i \subseteq D_j$.

Let $E^E = |t_1^E| \ddagger |t_2^E| \ddagger \dots \ddagger |t_k^E|$ be the elementary array expression of an OAT. These elementary arrays are the generators of the lattice structure. The other arrays of the lattice can be easily found by application of the $\circ$ product on all pairs of elementary arrays.

The general procedure, using array theory, that obtains the descriptions of the intensions of all formal concepts is the following:

1. Transform the OAT into an elementary array expression. Create ARRAYLIST with all elementary arrays (delete repeated arrays, if any).
2. For $|t_i^E|$ and $|t_j^E|$ belonging to ARRAYLIST perform,

$$|t_i^E| \circ |t_j^E| \quad \forall i, j$$

Add new arrays to ARRAYLIST. Some of the new arrays will be project-ars.

3. Repeat step 2 until no new arrays are found.

The final list is formed by elementary arrays and project-ars that describe the intensions of all formal concepts. These arrays are called concept arrays and are represented by $|t_y^C|$.

In order to find all the formal concepts of a context it is not necessary to know both the intension and the extension of a concept. The array theory presented above, finds the descriptions of the intensions of all concepts. These concept arrays are the descriptions of the formal concepts.

$$(D_y, R_y = \{g_y, \dots, b_y, a_y\}) \rightsquigarrow |t_y^C|$$

The partial order relation $\preceq$ allows the construction of the lattice structure. Given two concept arrays $|t_y^C|$ and $|t_x^C|$,

$$|t_y^C| \preceq |t_x^C| \quad \text{if and only if} \quad |t_y^C| \circ |t_x^C| = |t_y^C|$$

6 Conclusion

The proposed array algebra does not handle raw data, it handles declarative descriptions of the data. Declarative expressions from a multivalued OAT can be obtained using arrays and declarative expressions can be transformed by application of algebraic techniques.

The algebra of arrays allows the description of an arbitrary OAT by means of an array expression E_i that describes the same partial reality. These array expressions are not unique. The elementary array expression and the all-prime-ar expression have been singled out.

The n th order project-ars have been introduced in a two folded manner, as a zero array of degree n , and as a description of a reality with missing attributes. The degree of the zero array depends on the number of $\vee$ components in the array.

The starting point of Wille's concept lattice are the formal concepts. Both the intension and the extension must be known in order to have a concept. When all concepts are known, the construction of the lattice is a simple matter. The array theory introduced here, finds the concept arrays directly from the elementary array expression describing the OAT. That is, the array theory only finds the description of the intensional part of the concept. There is no need to know neither the identification of the objects, nor which objects are equal in the sense that they are described by the same set of attribute values. Furthermore,

the technique developed here is independent of the number of attributes and the number of values of each attribute. Multivalued tables and binary OAT's are treated similarly.

An important point to stress here is the semantic structure associated to the concepts: the objects in domain D_i have all attribute values of R_i . The concept lattice is constructed on this semantic structure. The objects D_i of the formal concepts, that is, the extensions are secondary to this semantic structure.

Finally, it should be mentioned that this array algebra has a dual version, the co-array algebra that has been introduced in [8] and should be further studied.

Acknowledgements. This work has been supported by the Universitat de les Illes Balears through the UIB 2003/11 project.

References

1. Michalski, R.S.: A Theory and Methodology of Inductive Learning. *Artificial Intelligence* **20** (1983) 111–161.
2. Quinlan, J. R.: Induction of Decision Trees. *Machine Learning* **1** (1986) 81–106.
3. Pawlak, Z.: *Rough Sets: Theoretical Aspects of Reasoning About Data*. Kluwer Academic Publisher (1991).
4. Bazan, J. and Skowron, A. and Synak, P.: Discovery of Decision Rules from Experimental Data. *Proceedings of the Third International Workshop on Rough Sets and Soft Computing* (1994) 346–355.
5. Miró, J. and Miró-Julià, J.: Uncertainty and Inference through Approximate Sets. *Uncertainty in Intelligent Systems*. North Holland (1993) 203–214.
6. Ganter, B. and Wille, R.: *Formal Concept Analysis. Mathematical Foundations*. Springer-Verlag (1999).
7. Fiol, Gabriel and Miró Nicolau, José and Miró-Julià, José: A New Perspective in the Inductive Acquisition of Knowledge from Examples. *Lecture Notes in Computer Science* **682** (1992) 219–228.
8. Miró-Julià, M.: *A Contribution to Multivalued Systems*. Ph.D. thesis. Universitat de les Illes Balears (2000).
9. Miró, J. and Miró-Julià, M.: A Numerical Computation for Declarative Expressions. *Lecture Notes in Computer Science* **1333** (1997) 236–251.
10. Miró, J. and Miró-Julià, M.: Equality of Functions in CAST. *Lecture Notes in Computer Science* **1030** (1995) 129–136.
11. Miró-Julià, M. and Miró, J.: Transformation of Array Expressions. *Proceedings of the Second IASTED International Conference. Artificial Intelligence and Applications* (2002) 273–278.
12. Wille, R.: Restructuring Lattice Theory: an Approach based on Hierarchies of Concepts. *Ordered Sets*, Reidel Publishing Company (1982) 445–470.

Invariants and Symmetries among Adaptive Agents

Germano Resconi

Department of Mathematics and Physics, Catholic University,
Via Trieste 17, 25100 Brescia, Italy
`resconi@numerica.it`

Abstract. In this paper we present agents and systems at different order of complexity. At the first order we have agents which acts are transformations from one state to another. At the second order the actions of the agents transform one context with rules and data to another context with other rules and data. In the transformation of the second order the new rules and the new data are projected in the new context. Rules of the new context must be adapted to the new rules obtained by the transformation. The adaptation process can be obtained by a close loop control inside the possible set of the rules included in the new context. Close loop control at the second order can be obtained by the second order action of the agents. Invariants and symmetry are possible when we move from one context to another. A set of contexts can be transformed in another set of contexts by a third order action of a meta-agent. In conclusion a hierarchical tower of agents and meta agents can be built in a way to generate complex transformations.

1 Introduction

We Know that a system or agent is $S^1 = [M(S^0), R(S^0)]$ where M is a set of objects S^0 and R is a set of relations among the objects. This is the system or agent of type one or system at the level one. In spite of a big success of the system model, we argue that more high level of systems or agents exists. System of the type two is $S^2 = [M(S^1), R(S^1)]$ where the object in $M(S^1)$ is a set of systems of the type one and $R(S^1)$ is a set of relations among the systems S^1 . For a system S^1 every relation R can be represented as a set of triples or statements. Three parts compose every triple. The first part is the initial term or source term, the second part is the property or relation and the third is the final term or value. We denote the set of initial terms as domain and the set of the final terms the range. When we transform domain and range, we obtain a set of triples that form a new relation R' that is the symmetric image of the relation R . An invariant is attached at every set of relations one symmetric image of the others. Because the domain and the range of R' are the transformation of the domain and range of R , for the relation R' we have a pre condition on the domain and a post condition on the range. When H transform one system A in another B where B has the domain and range transformed in the same way

in this case H is a homomorphism or an isomorphism. In general a relation in S^2 transform one triples or statements in another. The new set of statements generates a relation R' that is not always the symmetric image of the relation R i.e. R' cannot be obtained by a transformation of R . In this case the objects are not one the symmetric image of the other. Every process to restore the symmetry among the objects is denoted a compensation or adaptation process. The novelty in the system of the type two is the symmetry property among the objects and the possibility to restore the symmetry when is broken. With the system of the type two we can model sequential abstract state machines. In this case every object in S^2 is an abstract state machine where the internal relations are the semantic of the machine. The relation $R(S1)$ generate a semantic dynamic among the state machines. When in the dynamic we loss the original semantic so we loss the original symmetry, with an adaptation process we can restore the symmetry among abstract state machines. Before any implementation of the state machines, we can implement a system of type two or sequential abstract state machines in a way to explore the range of the symmetric image of the abstract state machines. Evolutionary autonomous agents can be represented by systems of the type two. In S^2 the objects or systems of the type one are agents. Among the agents we can have a symmetric image relation. Symmetric evolution of one agent means that one agent will be transformed, in time, in another agent that is its symmetric image. Adaptation among agents, transform non-symmetric agents into symmetric. Because the environment, where are locate the agents is another object in S^2 every agent can be a symmetric image of the environment or we can adapt the agents in a way to become a symmetric image of the environment. Close lop control at different orders can be implemented.

2 Adaptation Agent and Its Action

2.1 First Order Agent

In this chapter we present the fundamental mathematical and cognitive scheme where we locate the Adaptation Agent. We introduce the context definition at different orders, the transformations from one context to another and the adaptation processes by which we can establish a symmetric relation between the rules inside the different contexts. This chapter is not a completely abstract chapter but has the aim to introduce the lector to the field of the Adaptation Agent. When we have only one context every adaptation is impossible. In fact for the definition every context is make by variables and rules, where the rules are fixed inside the context. The adaptation principle has is source in the change of the rules, so because we have only one context and every change of the context is forbidden, we cannot have any adaptation process. In conclusion when we have only one context we have only one family of rules that cannot change. Every automatic system or program for a computer is an agent of the first order. For a formal description of the rules, is useful to describe the rules by an elementary semantic unity in this way Where IN symbolically represent the domain of the rule , P_1 the rule and OUT the range of the rule. In the table 1 we denoted the

Table 1. Semantic Unity

Statements	Resource	Property	Value
S_1	IN	P_1	OUT

Table 2. Example of Semantic Unity for contract net protocol

Statements	Resource	Property	Value
S_1	Manag. for proposal	P_1	Inter.Agent Send prop.

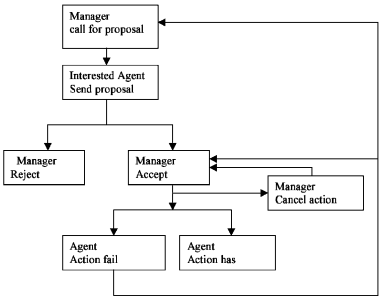


Fig. 1. Communication rules between a manager and a group of agents

domain as the Resource of the data that the agent use for the action. The rule becomes a property of the agent. So P_1 is the characteristic or the property of the agent. The range is the value of the action so OUT is denoted the value.

An example of rule at the first order is the FIPA **contract net protocol** that we show in Fig. 1

In this contract net protocol the domain and the range of the rule are equal and are given by the elements $V_1 = V_2 = \{ \text{"Manager call for proposal", "Interested agent send proposal", "Manager reject proposal", "Manager accept proposal", "Manager cancel proposal", "Agent give failure action", "Agent give success action"} \}$. In Fig. 1 we show the relation that exist between the actions of two agents, the Manager and the interested agents. The manager call for a proposal and a group of interested agents send a proposal to the manager. He can reject or accept the proposal. When the manager accept the proposal, the proposal can be cancelled with a cancelled action of the manager or because the agent fail in the realisation of the project. In any case the agent can send to the manager a new proposal or can ask a new accepted action from the manager. The rule of communications between two agents in Fig. 1 shows the context where we operate. The context is only one context and the Adaptive Agent cannot work because he must stay in the fixed rule of the context. One of the possible semantic unities can be see in Table 2.

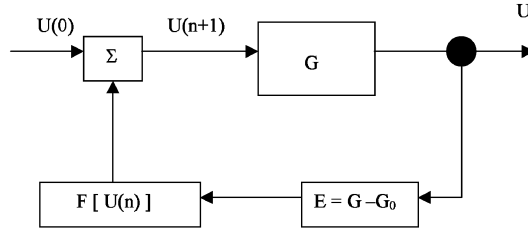


Fig. 2. Close loop control as a system at the first order

(2.0) Example. Close loop control or PID proportional integral derivative control as a first order control.

$$\frac{dU}{dt} = k_1 \frac{dE}{dt} + k_2 E \quad (2.1)$$

where E is the error and U is the control action. For the discrete system the previous equation can write in this way

$$U(n+1) - U(n) = K_1[E(n+1) - E(n)] + K_2 E(n)$$

When we write $K_1[E(n+1) - E(n)] + K_2 E(n) = F(U(n))$ we have

$$U(n+1) = F[U(n)] + U(n) \quad (2.2)$$

Fig. 2 shows this in a graphic way.

Because in the close loop control we have a relation between the set of inputs, at any step n the input move from one input to another. When the function $F[U(n)] = 0$, the error E and its variation go to zero. In this case the close loop becomes an open loop and the input ,output value of G does not depend on step n and the output has no error. The zero order system G_0 is the reference and the adaptation of the system to the value G_0 is a system of the first order with control.

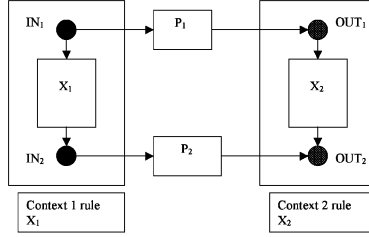
2.2 Second Order Action for Adaptive Agent

When we separate the domain of the knowledge in parts or contexts, we have the possibility to express the knowledge by different type of rules. Inside at every context we have fixed rules, but when we move from one context to another the rule can change also in an important way. Every agent before to use the rule for its action must be have knowledge on the context where it is locates. With the context he can know the rule and can give its action. Agents are divided in two orders. At the first order we have agent that use the rules, at the second order we have agents that change the rules- The semantic unity of agents of the second order or Adaptive Agent is given by the statements.

Every semantic unity of the agent of the second order change the rule X_1 in the first context into the rule X_2 in the second context. To simplify the

Table 3. a Semantic unity for agent at the second order

Statements	Resource	Property	Value
S_1	IN_1	X_1	IN_2
S_2	IN_1	P_1	OUT_1
S_3	IN_2	P_2	OUT_2
S_4	OUT_1	X_2	OUT_2

**Fig. 3.** Description of the semantic unity for the agent of the second order that change the rule X_1 inside the context 1 to the rule X_2 inside the context 2.

notations, at the proposition “Agent of the second order” we substitute the symbol “ $Agent^2$ ”.

The rules P_1 and P_2 are the instruments by which the $Agent^2$ change the rule X_1 into the rule X_2 . The semantic unity for $Agent^2$ can be graphically represented Fig. 3.

In Fig. 3 we represent in a graphic way the statement that we show in table 3. When we look the statement in the table 3, we remark that the S_1 and the S_2 have the same source IN_1 in the context 1 and that the statement S_3 and S_4 have the same value OUT_2 in the context 2. Because the statements can be represent in this way

$$S_1 \rightarrow P_1 \quad IN_1 = OUT_1, \quad S_2 \rightarrow X_1 \quad IN_1 = IN_2, \quad S_3 \rightarrow P_2 \quad IN_2 = OUT_2, \quad S_4 \rightarrow X_2 \quad OUT_1 = OUT_2$$

We have the relations among the properties

$$P_2 IN_2 = X_2 OUT_1 \text{ or } P_2 X_1 IN_1 = X_2 P_1 IN_1 \quad (2.3)$$

The relation 2.3 is the internal coherence for the Adaptive Agent or “ $Agent^2$ ”. When the relation 2.3 is false the adaptive agent is ambiguous or fuzzy. In the ambiguous case we have two different and conflicting representation of the same rule. For OUT_2 we have two different ranges. One external (Ex) that came from the first context by P_1 , P_2 , the other internal (In) generate by X_2 .

In Fig. 4 we show the two ranges. The second order of agent can be represented also by the RDM (Rapid Domain Modelling) as in the Fig 5.

(2.3) Example. Given the family of functions $OUT_2 = X_2[OUT_1, Q]$ where Q is a set of parameters. When we change the parameters Q , we change the function X_2 . The family of functions X_2 can be solutions of a differential equation.

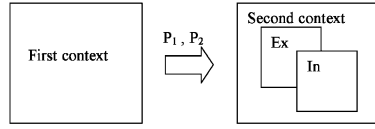


Fig. 4. Ranges of OUT_1 one External from the first context by P_1, P_2 and the Internal by X_2 .

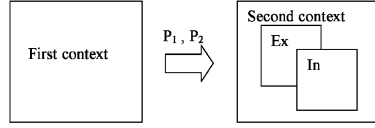


Fig. 5. Description of the semantic unity for the agent of the second order by RDM system.

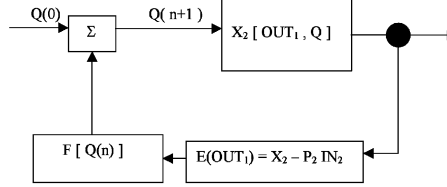


Fig. 6. Close loop control where the action of the control is the parameter Q and the loop depend on the value $OUT_1 = P_1 IN_1$.

Because we know the value OUT_2 obtained by the expression $OUT_2 = P_2 IN_2$ where IN_2 is in the first context, we can create a PID where the action of the control $U(n)$ is the parameter $Q(n)$ and $F[Q(n)] = K_1[E(n+1) - E(n)] + K_2E(n)$ where $E(OUT_1) = X_2(OUT_1, Q) - P_2 IN_2$. So we have the close loop control in Fig. 6.

In the discrete control PID we have that the error that we control is

$$F = K_1[E(n+1) - E(n)] + K_2E(n)$$

When we change the context we can compute a new error at the step n and at the step $n+1$ by this second order of agent

In the second context of the Fig. 7 the operator A is

$$A(U) = -\frac{dR^2}{dU}$$

where

$$R(U) = \frac{kF(U)}{\sqrt{1 + F(U)^2}} \text{ and } F(U) = K_1(E(n+1) - E(n)) + K_2E(n)$$

We can prove that $A(U)$ is zero when $F(U)$ is zero and that the close loop in the second context converge always.

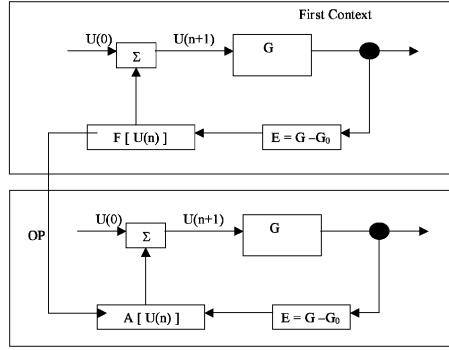


Fig. 7. Second order of close loop control. The operator P is the same for every step n .

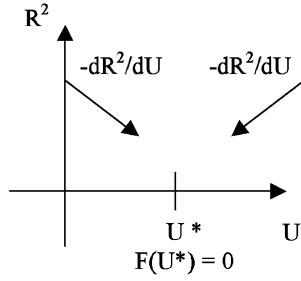


Fig. 8. Convergence of the operator A in the second context

2.3 Different Types of Actions for the Agent

First type or identical transformation. In the first type of the action the *Agent*² change the relation X^1 is the same in the context 1 and in the context 2. So $X_1 = X_2$ so the equation 2.3 can write in this way

$$P_2 X_1 I N_1 = X_1 P_1 I N_1 \quad (2.4)$$

To obtain the same rule X_1 in the second context the *Agent*² must balance the rules P_2 and P_1 in a way to have the same rule X_1 . When in every context we have the same rule the contexts are connected by a symmetry principle. For example the Newton law $F = ma$, where F is the vector of the force, m is the mass of the body and a is the vector of the acceleration, is valid in every context when the velocity is far from the velocity of the light.

Second type or transformation between similar rules. When the Adaptive Agent use the rules $P_1 = P_2 = P$ the rule X_2 is not equal to the relation X_1 but is similar. Every context sent to the other its similar image. For the relations 2.1 we have

$$P_1 X_1 = X_2 P_1 \quad (2.5)$$

Table 4. Action of the *Agent*² that connect similar rules between contexts

Statements	Resource	Property	Value
S_1	IN_1	X_1	IN_2
S_2	IN_1	P	OUT_1
S_3	IN_2	P	OUT_2
S_4	OUT_1	X_2	OUT_2

With the statement image the action of the agent of the second order is given in the table 4.

(2.5) Definition. A rule X_2 is similar to a rule X_1 when exist a transformation P for which

$$X_2OUT_1 = PIN_2 \quad (2.6)$$

(2.6) Property. The rule X_2 obtained by the second type of transformation is not always equal to the rule X_1 but is only similar.

In fact the rule X_2 can write in this way

$$X_2 : PIN_1 \rightarrow PIN_2 \text{ or } X_2 : PIN_1 \rightarrow PX_1IN_1 \text{ and } X_1 : IN_1 \rightarrow IN_2$$

given PIN_1 that belong to the domain of X_1 , the rule X_1 give the result X_1PIN_1 that can be different from PX_1IN obtained by X_2 . In conclusion only when P and X_1 commute we came back to the identical transformation in the first type 2.5 of transformations. We have two different aspects of the second type of transformation. The first aspect is relate to the situation where P is one-to one and onto a second aspect when P is only onto but is not one to one.

We can see that for one to one relation P the paths inside the relation X_1 are one to one to the paths in the relation X_2 . When P is onto the paths inside the relation X_1 are onto the paths in the relation X_2 .This is the traditional idea of isomorphism and the homomorphism. The difference is in the adaptation process by which X_2 can become similar to X_1 with a feedback process or other processes.

Third type or transformations between rules that are not similar.

When the Adaptive Agent use the rules P_1 and P_2 where P_1 is different from P_2 the rule X_2 is not equal or similar to the relation X_1 . Every context send to the other an image that is not equal or similar. Also if P_1 is different from P_2 , we can write again the relations 2.3 in its general form

$$P_2X_1 = X_2P_1 \quad (2.7)$$

With the statement image the action of the agent of the second order is given in the table 5.

The important property of the third type of transformation is that we can change (compensation) the rule X_2 in a way that X_1 is similar to X_2 or X_1

Table 5. Action of the *Agent*² that connect rules X_1 and X_2 that are not similar

Statements	Resource	Property	Value
S_1	IN_1	X_1	IN_2
S_2	IN_1	P_1	OUT_1
S_3	IN_2	P_2	OUT_2
S_4	OUT_1	X_2	OUT_2

Table 6. Action of the *Agent*² that connect similar rules between rule X_1 and rule X_2G . Where X_2G is the compensate rule of X_2 by the compensate term G

Statements	Resource	Property	Value
S_1	IN_1	X_1	IN_2
S_2	IN_1	P_1	OUT_1
S_3	IN_2	P_2	OUT_2
S_4	OUT_1	X_2G	OUT_2

Table 7. Action of the *Agent*² that connect similar rules between rule X^1 and rule HX_1 . Where HX_1 is the compensate rule of X_2 by the compensate term H

Statements	Resource	Property	Value
S_1	IN_1	X_1	IN_2
S_2	IN_1	P_2	OUT_1
S_3	IN_2	P_2	OUT_2
S_4	OUT_1	X_2G	OUT_2

$\approx X'_2$. We can also change the rule X_1 in a way that the new rule is similar to the rule X_2 . or $X'_1 \approx X_2$. In fact the (2.7) can write in this way

$$P_2X_1 = X_2GP_2 \text{ where } P_1 = GP_2 \text{ and } G = P_1P_2 - 1 \quad (2.8)$$

With the statement image of the action of the second order we have Table 6.

For the second type action 2.4.2 we have that X_1 is similar to X_2G or $X_1 \approx X_2G$ Where G is the compensative rule which action change the rule X_2 . But for (6) we can also write this expression

$$P_1HX_1 = X_2P_1 \text{ where } P_2 = HP_1 \text{ or } H = P_2P_1 - 1 \quad (2.9)$$

With the statement image of the action of the second order we have Table 7.

For the second type action 2.4.2 we have that HX_1 is similar to X_2 or $HX_1 \approx X_2$.

Explanation of the compensatory rules H and G by the internal and external rules. When P_2 is different from P_1 , the action or rule in the second context is divided in an internal action and in an external action. When in a second context the agent receive a message he make an internal elaboration and after an external elaboration. The internal elaboration is G and the external elaboration is X_2 . The total computation X_2G is similar, see 2.9 to the rule X_1 but the individual computation G or X_2 are not similar to X_1 . In conclusion the similarity with

Table 8. Action of the *Agent*² that connect similar rules between rule X^1 and rule HX_1 . Where HX_1 is the compensate rule of X_2 by the compensate term H

Statements	Resource	Property	Value
S_1	IN_{11}	X_1	IN_{21}
S_2	IN_{11}	P_1	OUT_{11}
S_3	IN_{11}	A_1	IN_{12}
S_4	IN_{21}	P_2	OUT_{21}
S_5	IN_{21}	A_3	IN_{22}
S_6	OUT_{11}	X_2	OUT_{21}
S_7	OUT_{11}	A_2	OUT_{12}
S_8	IN_{12}	X_3	IN_{22}
S_9	IN_{12}	P_3	OUT_{12}
S_{10}	IN_{22}	P_4	OUT_{22}
S_{11}	OUT_{21}	A_4	OUT_{22}
S_{12}	OUT_{12}	X_4	OUT_{22}

X_1 is realised by the internal computation G that in this way compensate the non-similarity of X_2 with X_1 . We can repeat all the previous idea of internal and external rule with the compensatory rule H (brain) that is the internal rule and X_1 as the external rule. In this case we have that HX_1 is similar to the rule X_2 see (2.9).

2.4 Third Order Actions in Adaptive Agent

When we separate the domain of the knowledge in cognitive parts, we can separate the knowledge in rules of the first order, rules of the second order and rules at the third order. Because at every rule we associate an action of one agent, we can have actions at the first order, at the second order and at the third order. When we separate the knowledge in different contexts, we have context of the first order that include rules or methods at the first order, contexts at the second orders that include rules at the second order and context at the third order that include rules at the third order.

To simplify the notations, at the proposition “Agent of the third order” we substitute the symbol “*Agent*³”.

The rules A_1, A_2, A_3, A_4 are the instruments by which the *Agent*³ change the rule X_1 into the rule X_4 . Because

$$S_7 \rightarrow A_2 \text{ } OUT_{11} = OUT_{12}, S_9 \rightarrow P_3 \text{ } IN_{12} = OUT_{12}, S_4 \rightarrow P_2 \text{ } IN_{21} = OUT_{21} \\ S_6 \rightarrow X_2 \text{ } OUT_{11} = OUT_{21}, S_5 \rightarrow A_3 \text{ } IN_{21} = IN_{22}, S_8 \rightarrow X_3 \text{ } IN_{12} = IN_{22}.$$

For the previous statements we have that

$$P_3 IN_{12} = P_3 \text{ } IN_{12} = OUT_{12}, P_2 IN_{21} = X_2 \text{ } OUT_{11} = OUT_{21}, A_3 IN_{21} = X_3 \text{ } IN_{12} = IN_{22}$$

For the table 8 we have the three equations

$$A_2 P_1 = P_3, A_1 = OUT_{12} \quad P_2 X_1 = X_2, P_1 = OUT_{21}, \quad A_3 X_1 = X_3, A_1 = IN_{22} \quad (2.10)$$

We have also that the statement S_10, S_11, S_12 have the same value OUT_{22} in the context 2 of the second order. So we have

$$S_10 \rightarrow P_4 IN_{22} = OUT_{22}, S_{11} \rightarrow A_4 OUT_{21} = OUT_{22}, S_{12} \rightarrow X_4 OUT_{12} = IN_{22}.$$

For the previous statements we have

$$P_4 IN_{22} = A_4, OUT_{21} = X_4 OUT_{12}$$

For the equation 2.10 we have

$$P_4 A_3 X_1 = A_4 P_2 X_1 = X_4 A_2 P_1 \quad (2.11)$$

The relations (2.8) are the internal coherence for the three Adaptive Agent or “*Agent*²” included in the Adaptive Agent Action of the order three “*Agent*³”. The three agents *Agent*² change always IN_{11} has initial object. The relation (2.10) is the internal coherence for Adaptive Agent of the order three or “*Agent*³”. When one of the equations (2.10) or (2.11) is false the adaptive agent is ambiguous or fuzzy.

3 Conclusion

In this paper we present agents or systems at different order. From the order two agents before generate its actions or transformation must be control its coherence or symmetry. After the action is possible. In the ordinary agent action is free and no a priori conditions are necessary.

References

1. Resconi Germano, Gillian Hill,[1996] The Language of General Systems Logical Theory: a Categorical View, [European Congress on Systems Science Rome 1–4, October 1996]
2. C.Ratray,G.Resconi,G.Hill, GSLT and software Development Process, Eleventh International Conference on Mathematical and Computer Modelling and Scientific Computing,[March 31–April 3,1997,Georgetown University, Washington D.C.]
3. Mignani R. E.Pessa,G.Resconi, [1993]Commutative diagrams and tensor calculus in Riemann spaces, [Il Nuovo Cimento 108B(12) December 1993]
4. Resconi G. M.Jessel, [1986] A General System Logical Theory, [International Journal of General Systems 12: 159–182]
5. Resconi, G, Ratray, C and Hill, G, [1998] The Language of General Systems Logical Theory (GSLT), [Int’l Journal of General Systems]
6. Mariusz Nowostawski , Martin Puvis , Stephen Cranefield, A layered Approach for Modelling Agent Conversations, The information Science Discussion paper series , [Number 2001/05 , March 2001, ISSN 1172–6024]
7. Raul Izquierdo Castanedo , RDM: Arquitectura Software para el Modelado de Dominios en Sistemas Informatico Tesis Doctoral , [Universidad De Oviedo, Junio de 2002]

Generalizing Programs via Subsumption^{*}

Miguel A. Gutiérrez-Naranjo, José A. Alonso-Jiménez, and
Joaquín Borrego-Díaz

Dept. of Computer Science and Artificial Intelligence
University of Seville
{magutier,jalonso,jborrego}@us.es

Abstract. In this paper we present a class of operators for Machine Learning based on Logic Programming which represents a *characterization* of the subsumption relation in the following sense: The clause C_1 subsumes the clause C_2 iff C_1 can be reached from C_2 by applying these operators. We give a formalization of the closeness among clauses based on these operators and an algorithm to compute it as well as a bound for a quick estimation. We extend the operator to programs and we also get a characterization of the subsumption between programs. Finally, a weak metric is presented to compute the closeness among programs based on subsumption.

1 Introduction

In a Machine Learning system based on clausal logic, the main operation lies on applying an operator to one or more clauses with the hope that the new clauses give a better classification for the training set. This generalization must fit into some relation of order on clauses or sets of clauses. The usual orders are the subsumption order, denoted by $\succeq$, and the implication order $\models$.

Subsumption was presented by G. Plotkin [9]. In his study about the lattice structure induced by this relation on the set of clauses, he proved the existence of the *least general generalization* of two clauses under subsumption and defined the *least generalization under relative subsumption*. Both techniques are the basis of successful learning systems on real-life problems. Later, different classes of operators on clauses, the so-called refinement operators, were studied by Shapiro [11], Laird [5] and van der Laag and Nienhuys-Cheng [13] among others. In their works, the emphasis is put on the specialization operators, which are operators such that the obtained clause is implied or subsumed by the original clause, and the generalization operators are considered the *dual* of the first ones.

In this paper we present new results and algorithms about the generalization of clauses and logic programs via subsumption. We propose *new* generalization operators for clausal learning, the Learning Operators under Subsumption which represent a *characterization* by operators of the subsumption relation between

^{*} Work partially supported by project TIC 2000-1368-C03-0 (Ministry of Science and Technology, Spain) and the project TIC-137 of the *Plan Andaluz de Investigación*.

clauses in the following sense: If C_1 and C_2 are clauses, C_1 subsumes C_2 if and only if there exists a finite sequence (a *chain*) of LOS $\{\Delta_1/x_1\}, \dots, \{\Delta_n/x_n\}$ such that $C_1 = C_2\{\Delta_1/x_1\} \dots \{\Delta_n/x_n\}$. If C_1 subsumes C_2 , we know that the set of chains of LOS from C_2 to C_1 is not empty, but in general the set has more than one element.

The existence of a non-empty set of chains gives us the idea for a formalization of *closeness* among clauses as the length of the shortest chain from C_2 to C_1 , if C_1 subsumes C_2 , and infinity otherwise.

This mapping, which will be denoted by dc , is the algebraic expression of the subsumption order: for every pair of clauses, C_1 and C_2 , C_1 subsumes C_2 if and only if $dc(C_2, C_1)$ is finite. Since the subsumption order is not symmetric, the mapping dc is not either. Therefore dc is not a metric, but a *quasi-metric*.

Finally, dc is computable. We give in this paper an algorithm which calculates the quasi-distance between two clauses and present a bound which allows to estimate the closeness between clauses under the hypothesis of subsumption. This algorithm and estimation provides useful tools for the design of new learning systems which use quasi metrics to compute closeness.

In the second part of the paper, we extend the study to programs. We define a new class of operators, the *composed LOS*, which act on the set of programs and they also represent a characterization of the subsumption relation between programs. Analogously to the clausal case, the minimum of the length of the chains of operators between two programs is the basis of a weak metric to quantify the closeness between programs. This weak metric has been experimentally checked and can be added to existing systems or used to design new ones.

2 Preliminaries

From now on, we will consider some fixed first-order language $\mathcal{L}$ with at least one function symbol. *Var*, *Term* and *Lit* are, respectively, the sets of variables, terms and literals of $\mathcal{L}$. A *clause* is a finite set of literals, a *program* is a non-empty finite set of non-empty clauses, $\mathbb{C}$ is the set of all clauses and $\mathbb{P}$ is the set of all programs. A definite program is a program where each clause contains one positive and zero or more negative literals. As usual, T_P will denote the immediate consequence operator of the program P .

A *substitution* is a mapping $\theta : S \rightarrow Term$ where S is a finite set of variables such that $(\forall x \in S)[x \neq \theta(x)]$. We will use the usual notation $\theta = \{x/t : x \in S\}$, where $t = \theta(x)$, $Dom(\theta)$ for the set S and $Ran(\theta) = \cup\{Var(t) : x/t \in \theta\}$. A pair x/t is called a *binding*. If A is a set, then $|A|$ is the cardinal of A and $\mathcal{P}A$ its power set. We will denote by $|\theta|$ the number of bindings of the substitution θ . The clause C *subsumes* the clause D , $C \succeq D$, iff there exists a substitution θ such that $C\theta \subseteq D$. A *position* is a non-empty finite sequence of positive integers. Let $\mathbb{N}^+$ denote the set of all positions. If $t = f(t_1, \dots, t_n)$ is an atom or a term, t_i is the term at position i in t and the term at position $i \hat{u}$ in t is s if s is at position u in t_i . Two positions u and v are *independent* if u is not a prefix of v and vice versa. A set of positions P is *independent* if it is a pairwise independent

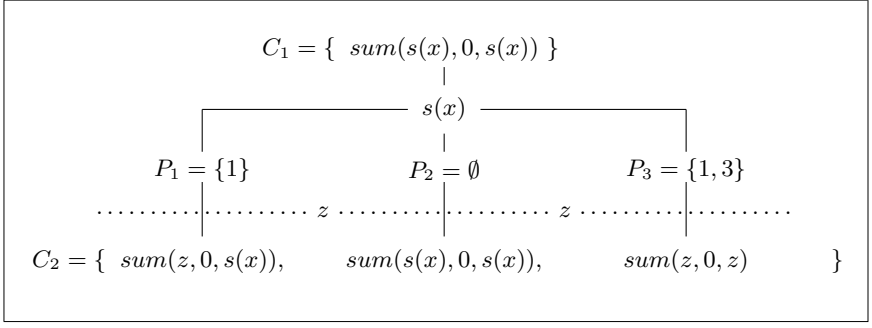


Fig. 1. Example of generalization

set of positions. The set of all positions of the term t in L will be denoted by $Pos(L, t)$. If t is a term (resp. an atom), we will denote by $t[u \leftarrow s]$ the term (resp. the atom) obtained by grafting the term s in t at position u and, if L is a literal, we will write $L[P \leftarrow s]$ for the literal obtained by grafting the term s in L at the independent set of positions P .

3 The Operators

In the generalization process, when a program P is too specific, we replace it by P' with the hope that P' covers the examples better than P . The step from P to P' is usually done by applying an operator to some clause C of P . These operators can be defined as mappings from $\mathbb{C}$ to $\mathbb{C}$, where $\mathbb{C}$ is the set of clauses of the language. Before giving the definition of the operator, we will give some intuition with an example.

Consider the one-literal clause $C_1 = \{L\}$ with $L = \text{sum}(s(x), 0, s(x))$. In order to generalize it with respect to the subsumption order, we have to obtain a new clause C_2 such that there exists a substitution θ verifying $C_2\theta \subseteq C_1$. For that, we firstly choose a term t in L , say $t = s(x)$, then we choose several subsets of $Pos(L, t)$, e.g. $P_1 = \{1\}, P_2 = \emptyset, P_3 = \{1, 3\}$ and a variable not occurring in L , say z , and finally we build the clause $C_2 = \{L[P_i \leftarrow z] \mid i = 1, 2, 3\} = \{\text{sum}(z, 0, s(x)), \text{sum}(s(x), 0, s(x)), \text{sum}(z, 0, z)\}$. Obviously $\theta = \{z/s(x)\}$ satisfies $C_2\theta \subseteq C_1$ (see Fig. 1). If the clause has several literals, for example, $C_1 = \{L_1, L_2, L_3\}$, with $L_1 = \text{num}(s(x))$, $L_2 = \text{less\_than}(0, s(x))$ and $L_3 = \text{less\_than}(s(x), s(s(x)))$, the operation is done with all literals simultaneously. First, the same term is chosen in every literal of C_1 , say $t = s(x)$. Then, for each literal $L_i \in C_1$, some subsets of $Pos(L_i, t)$ are chosen, e.g., $P_1^* = \{\emptyset, \{1\}\} \subseteq \mathcal{P}Pos(L_1, t)$, $P_2^* = \emptyset \subseteq \mathcal{P}Pos(L_2, t)$ and $P_3^* = \{\{1, 2 \cdot 1\}, \{1\}\} \subseteq \mathcal{P}Pos(L_3, t)$. After taking a variable which does not occur in C_1 , say z , we build the sets $L_1 \xrightarrow{P_1^*} \{\text{num}(s(x)), \text{num}(z)\}$, $L_2 \xrightarrow{P_2^*} \emptyset$ and $L_3 \xrightarrow{P_3^*} \{\text{less\_than}(z, s(z)), \text{less\_than}(z, s(s(x)))\}$. C_2 is the union of these sets, i.e., $C_2 = \{\text{num}(s(x)), \text{num}(z), \text{less\_than}(z, s(z)), \text{less\_than}(z, s(s(x)))\}$

and $C_2\{z/s(x)\} \subseteq C_1$. In our general description, we will begin with substitutions and grafts.

Definition 1. Let L be a literal and t a term. The set of positions P is called compatible with the pair $\langle L, t \rangle$ if $P \subseteq \text{Pos}(L, t)$.

Let P^* be a set whose elements are sets of positions. Let L be a literal and t a term. P^* is called compatible with the pair $\langle L, t \rangle$ if every element of P^* is compatible with $\langle L, t \rangle$

For example, if $L = \text{sum}(s(x), 0, s(x))$ and $t = s(x)$, then $P_1 = \{1\}$, $P_2 = \emptyset$, $P_3 = \{1, 3\}$ are compatible with $\langle L, t \rangle$ but $P_4 = \{1 \cdot 1, 2\}$, $P_5 = \{1, 4 \cdot 3\}$ are not. If $P_1^* = \{P_1, P_2, P_3\}$ and $P_2^* = \{P_2, P_4\}$, then P_1^* is compatible with $\langle L, t \rangle$ and P_2^* is not.

The next mappings are basic in the definition of our operators. As we saw in the example, the key is to settle a set of sets of positions for each literal, all them occupied by the same term. This one is done by the following mappings.

Definition 2. A mapping $\Delta : \text{Lit} \rightarrow \mathcal{PPN}^+$ is an assignment if there exists a term t such that, for every literal L , $\Delta(L)$ is compatible with the pair $\langle L, t \rangle$.

Note that the term t does *not* have to be unique, for example, consider the *identity* assignment $(\forall L \in \text{Lit})[\Delta(L) = \{\emptyset\}]$, the *empty* assignment $(\forall L \in \text{Lit})[\Delta(L) = \emptyset]$ or any mixture of both.

The assignments map a literal into a set of sets of positions. Each element of this set of positions will produce a literal, and the positions are the places where the new term is grafted. If $\Delta : \text{Lit} \rightarrow \mathcal{PPN}^+$ is an assignment of positions and s is a term, we will denote by $L\{\Delta(L)/s\}$ the set of literals, one for each element $P \in \Delta(L)$, obtained by grafting s in L at P . Formally $L\{\Delta(L)/s\} = \{L[P \leftarrow s] \mid P \in \Delta(L)\}$ For example, if $L = \text{sum}(s(x), 0, s(x))$, z is a variable, P_1^* is taken from the above example and Δ is an assignment such that $\Delta(L) = P_1^*$ then $L\{\Delta(L)/z\} = \{L[P \leftarrow z] \mid P \in \Delta(L)\} = \{L[P \leftarrow z] \mid P \in P_1^*\} = \{L[P_1 \leftarrow z]\}$, $L[P_2 \leftarrow z], L[P_3 \leftarrow z]\} = \{\text{sum}(z, 0, s(x)), \text{sum}(s(x), 0, s(x)), \text{sum}(z, 0, z)\}$ We can now define our Learning Operators under Subsumption¹.

Definition 3. Let Δ be an assignment and x a variable. The mapping

$$\begin{aligned} \{\Delta/x\} : \mathbb{C} &\longrightarrow \mathbb{C} \\ C &\mapsto C\{\Delta/x\} = \bigcup_{L \in C} L\{\Delta(L)/x\} \end{aligned}$$

is a Learning Operator under Subsumption (LOS) if for all literal L , if $\Delta(L) \neq \emptyset$ then $x \notin \text{Var}(L)$.

Turning back to a previous example, if $C = \{L_1, L_2, L_3\}$, with $L_1 = \text{num}(s(x))$, $L_2 = \text{less\_than}(0, s(x))$, $L_3 = \text{less\_than}(s(x), s(s(x)))$, and the assignment

$$\Delta(L) = \begin{cases} P_1^* = \{\emptyset, \{1\}\} & \text{if } L = L_1 \\ P_2^* = \emptyset & \text{if } L = L_2 \\ P_3^* = \{\{1, 2 \cdot 1\}, \{1\}\} & \text{if } L = L_3 \\ \emptyset & \text{otherwise} \end{cases}$$

¹ A preliminary version of these operators appeared in [2].

and considering z as the variable to be grafted, then $C\{\Delta/z\} = \{num(s(x)), num(z), less\_than(z, s(z)), less\_than(z, s(s(x)))\}$. These operators allow us to generalize a given clause and go up in the subsumption order on clauses as we see in the next theorem.

Proposition 1. *Let C be a clause and $\{\Delta/x\}$ a LOS. Then $C\{\Delta/x\} \succeq C$.*

The LOS define an operational definition of the subsumption relation. The last result states one way of the implication. The next one claims that all the learning based on subsumption of clauses can be carried out only by applying LOS.

Theorem 1. *Let C_1 and C_2 be two clauses such that $C_1 \succeq C_2$. Then there exists a finite sequence (a chain) $\{\Delta_1/x_1\}, \dots, \{\Delta_n/x_n\}$ of LOS such that*

$$C_1 = C_2\{\Delta_1/x_1\} \dots \{\Delta_n/x_n\}$$

For example, if we consider $C_1 = \{p(x_1, x_2)\}$ and $C_2 = \{p(x_2, f(x_1)), p(x_1, a)\}$ and the substitution $\theta = \{x_1/x_2, x_2/f(x_1)\}$. Then $C_1\theta \subseteq C_2$ holds and therefore $C_1 \succeq C_2$. Decomposing θ we can get $\sigma_1 = \{x_2/x_3\}$, $\sigma_2 = \{x_1/x_2\}$, $\sigma_3 = \{x_3/f(x_1)\}$ and $C_1\sigma_1\sigma_2\sigma_3 \subseteq C_2$ holds. Hence, considering the assignments

$$\begin{aligned} \Delta_1(p(x_2, f(x_1))) &= \{\{2\}\} & \text{and } \Delta_1(L) &= \emptyset \text{ if } L \neq p(x_2, f(x_1)) \\ \Delta_2(p(x_2, x_3)) &= \{\{1\}\} & \text{and } \Delta_2(L) &= \emptyset \text{ if } L \neq p(x_2, x_3) \\ \Delta_3(p(x_1, x_3)) &= \{\{2\}\} & \text{and } \Delta_3(L) &= \emptyset \text{ if } L \neq p(x_1, x_3) \end{aligned}$$

we have $C_1 = C_2\{\Delta_1/x_3\}\{\Delta_2/x_1\}\{\Delta_3/x_2\}$. Note that if we take the assignment $\Delta(p(x_1, a)) = \{\{2\}\}$; $\Delta(L) = \emptyset$ if $L \neq p(x_1, a)$, then $C_1 = C_2\{\Delta/x_2\}$ also holds.

4 A Quasi-metric Based on Subsumption

The operational characterization of the subsumption relation given in the previous section gives us a natural way of formalizing the *closeness* among clauses. As we have seen, if $C_1 \succeq C_2$ then there exists *at least* one chain of LOS from C_2 to C_1 and we can consider the length of the shortest chain from C_2 to C_1 . If C_1 does not subsume C_2 , we will think that C_1 cannot be reached from C_2 by applying LOS, so both clauses are separated by an infinite distance.

Definition 4. *A chain of LOS of length n ($n \geq 0$) from the clause C_2 to the clause C_1 is a finite sequence of n LOS $\{\Delta_1/x_1\}, \{\Delta_2/x_2\}, \dots, \{\Delta_n/x_n\}$ such that $C_1 = C_2\{\Delta_1/x_1\}\{\Delta_2/x_2\} \dots \{\Delta_n/x_n\}$. The set of all the chains from C_2 to C_1 will be denoted by $\mathbf{L}(C_2, C_1)$ and $|\mathcal{C}|$ will denote the length of the chain $\mathcal{C}$. We define the mapping $dc: \mathbb{C} \times \mathbb{C} \rightarrow [0, +\infty]$ as follows:*

$$dc(C_2, C_1) = \begin{cases} \min\{|\mathcal{C}| : \mathcal{C} \in \mathbf{L}(C_2, C_1)\} & \text{if } C_1 \succeq C_2 \\ +\infty & \text{otherwise} \end{cases}$$

The subsumption relation is not symmetric, so the mapping dc is not either. Instead of being a drawback, this property gives an algebraic characterization of the subsumption relation, since $C_1 \succeq C_2$ iff $dc(C_2, C_1) \neq +\infty$. Notice that a quasi-metric satisfies the conditions to be a metric, except for the condition of symmetry.

Definition 5. A quasi-metric on a set X is a mapping d from $X \times X$ to the non-negative reals (possibly including $+\infty$) satisfying: (1) $(\forall x \in X) d(x, x) = 0$, (2) $(\forall x, y, z \in X) d(x, z) \leq d(x, y) + d(y, z)$ and (3) $(\forall x, y \in X) [d(x, y) = d(y, x) = 0 \Rightarrow x = y]$

The next result states the computability of dc and provides an algorithm to compute it.

Theorem 2. dc is a computable quasi-metric.

Proof (Outline). Proving that dc is a quasi-metric is straightforward from the definition. The proof of the computability is split in several steps. Firstly, for each substitution θ we define the set of the splittings up:

$$Split(\theta) = \left\{ \sigma_1 \dots \sigma_n : \begin{array}{l} \sigma_i = \{x_i/t_i\} \quad x_i \notin Var(t_i) \\ (\forall z \in Dom(\theta)) [z\theta = z\sigma_1 \dots \sigma_n] \end{array} \right\}$$

with $length(\sigma_1 \dots \sigma_n) = n$ and $weight(\theta) = \min\{length(\Sigma) \mid \Sigma \in Split(\theta)\}$. The next equivalence holds

$$dc(C_2, C_1) = \begin{cases} 0 & \text{if } C_1 = C_2 \\ 1 & \text{if } C_1 \neq C_2 \text{ and } C_1 \subseteq C_2 \\ \min\{weight(\theta) \mid C_1\theta \subseteq C_2\} & \text{if } C_1 \supseteq C_2 \text{ and } C_1 \not\subseteq C_2 \\ +\infty & \text{if } C_1 \not\supseteq C_2 \end{cases}$$

We can decide if $C_1 \supseteq C_2$ and, if it holds, we can get the finite set of all θ such that $C_1\theta \subseteq C_2$, so to conclude the theorem we have to give an algorithm which computes $weight(\theta)$ for each θ . The Fig. 2 shows a non-deterministic algorithm which generates elements of $Split(\theta)$. The algorithm finishes and for all $\Sigma \in Split(\theta)$ it outputs $\Sigma^* \in Split(\theta)$ verifying $length(\Sigma^*) \leq length(\Sigma)$.

The previous theorem provides a method for computing dc , but deciding whether two clauses are related by subsumption is an NP-complete problem [1], so, from a practical point of view we need a quick estimation of the quasi-metric before deciding the subsumption. The next result settles an upper and lower bounds for the quasi-metric under the assumption of subsumption.

Theorem 3. Let C_1 and C_2 be two clauses such that $C_1 \not\subseteq C_2$. If $C_1 \supseteq C_2$ then

$$|Var(C_1) - Var(C_2)| \leq dc(C_2, C_1) \leq \min\{2 \cdot |Var(C_1)|, |Var(C_1)| + |Var(C_2)|\}$$

Proof (Outline). For each θ such that $C_1\theta \subseteq C_2$, θ has at least $|Var(C_1) - Var(C_2)|$ bindings and we need at least one LOS for each binding, hence the first inequality holds. For the second one, if $C_1\theta \subseteq C_2$ then we can find n substitutions $\sigma_1, \dots, \sigma_n$ with $\sigma_1 = \{x_i/t_i\}$ and $x_i \notin Var(t_i)$ such that $C_1\sigma_1 \dots \sigma_n \subseteq C_2$ verifying $n = |\theta| + |Ran(\theta) \cap Dom(\theta)|$. The inequality holds since $Ran(\theta) \subseteq Var(C_2)$, $Dom(\theta) \subseteq Var(C_1)$ and $|\theta| \leq |Var(C_1)|$. If $C_1 = \{p(x_1, x_2)\}$, $C_2 = \{p(a, b)\}$ and $C_3 = \{p(f(x_1, x_2), f(x_2, x_1))\}$ then

$$\begin{aligned} dc(C_2, C_1) &= |Var(C_1) - Var(C_2)| = 2 \\ dc(C_3, C_1) &= \min\{2 \cdot |Var(C_1)|, |Var(C_1)| + |Var(C_2)|\} = 4 \end{aligned}$$

The above examples show that these bounds cannot be improved.

Input: A non-empty substitution θ
Output: An element of $Split(\theta)$
Set $\theta_0 = \theta$ and $U_0 = Dom(\theta) \cup Ran(\theta)$

Step 1:
 If θ_i is the empty substitution
 Then stop
 Otherwise: Consider $\theta_i = \{x_1/t_1, \dots, x_n/t_n\}$ and go to **Step 2**.

Step 2:
 If there exists $x_j \in Dom(\theta_i)$ such that $x_j \notin Ran(\theta_i)$
 Then for all $k \in \{1, \dots, j-1, j+1, \dots, n\}$ let t_k^* be a term
 such that $t^k = t_k^*\{x_j/t_j\}$ Set
 $\theta_{i+1} = \{x_1/t_1^*, \dots, x_{j-1}/t_{j-1}^*, x_{j+1}/t_{j+1}^*, \dots, x_n/t_n^*\}$
 $\sigma_{i+1} = \{x_j/t_j\}$
 $U_{i+1} = U_i$
 set i to $i+1$ and go to **Step 1**.
 Otherwise: Go to **Step 3**.

Step 3:
 In this case let z_i be a variable which does not belong to U_i and set
 $U_{i+1} = U_i \cup \{z_i\}$
 choose $j \in \{1, \dots, n\}$ y let T be a subterm of t_j such that T is not a variable
 belonging to U_{i+1} . Then, for all $k \in \{1, \dots, n\}$ let t_k^* be a term
 such that $t^k = t_k^*\{z/T\}$. Set
 $\theta_{i+1} = \{x_1/t_1^*, \dots, x_n/t_n^*\}$
 $\sigma_{i+1} = \{z/T\}$
 set i to $i+1$ and go to **Step 1**.

Fig. 2. Algorithm scheme to compute the subset of $Split(\theta)$

5 Programs

In this section we extend the study of subsumption to programs.

Definition 6. *The program P_2 subsumes the program P_1 , $P_2 \succeq P_1$, if there exists a mapping $F : P_1 \rightarrow P_2$ such that $F(C) \succeq C$, for all $C \in P_1$.*

In the case of definite programs, the subsumption is related to the semantics via the immediate consequence operator. The proof is adapted from [7].

Proposition 2. *Let P_1 and P_2 be two definite programs. Then $P_1 \succeq P_2$ if and only if for all interpretation I , $T_{P_2}(I) \subseteq T_{P_1}(I)$*

The operators for programs are sets of pairs assignment–variable. These operators represent a characterization for the subsumption relation between programs, as we will show below.

Definition 7. *A composed LOS is a finite set of pairs $\Theta = \{\Delta_1/x_1, \dots, \Delta_n/x_n\}$ where $\{\Delta_i/x_i\}$ is a LOS for all $i \in \{1, \dots, n\}$.*

For applying a composed LOS to a program we need an auxiliary mapping which associates one LOS to each clause of the program.

Definition 8. Let P be a program and Θ a composed LOS. An auxiliary mapping for applying (amfa) is a mapping $a : P \rightarrow \Theta$ such that for all clause $C \in P$, the clause $C\{a(C)\}$ is not the empty clause. The program $P_a\Theta = \{C\{a(C)\} \mid C \in P\}$ is the program obtained by applying Θ to P via the amfa a .

For example, consider the program² $P = \{C_1, C_2, C_3\}$ with

$$\begin{aligned} C_1 &= \text{sum}(0, s(s(0)), s(s(0))) \leftarrow \text{sum}(0, s(0), s(0)) \\ C_2 &= \text{sum}(s(x), s(0), s(z)) \leftarrow \text{sum}(s(0), 0, s(0)), \text{sum}(x, s(0), z) \\ C_3 &= \text{sum}(s(y), y, s(z)) \leftarrow \text{sum}(0, y, y), \text{sum}(y, y, z) \end{aligned}$$

and $\Theta = \{\Delta_1/x, \Delta_2/y, \Delta_3/x\}$ with

$$\begin{aligned} \Delta_1(L) &= \begin{cases} \{\{2, 3\}\} & \text{if } L = \text{sum}(0, s(s(0)), s(s(0))) \\ \emptyset & \text{otherwise} \end{cases} \\ \Delta_2(L) &= \begin{cases} \{\{2\}\} & \text{if } L \in \left\{ \begin{array}{l} \text{sum}(s(x), s(0), s(z)) \\ \neg \text{sum}(x, s(0), z) \end{array} \right\} \\ \emptyset & \text{otherwise} \end{cases} \\ \Delta_3(L) &= \begin{cases} \{\{1 \cdot 1\}\} & \text{if } L = \text{sum}(s(y), y, s(z)) \\ \{\{1\}\} & \text{if } L = \neg \text{sum}(y, y, z) \\ \emptyset & \text{otherwise} \end{cases} \end{aligned}$$

Consider the amfa $a : P \rightarrow \Theta$ such that $a(C_1) = \Delta_1/x$, $a(C_2) = \Delta_2/y$, $a(C_3) = \Delta_3/x$. Then

$$\begin{aligned} C_1\{a(C_1)\} &= C_1\{\Delta_1/x\} \equiv \text{sum}(0, x, x) \leftarrow \\ C_2\{a(C_2)\} &= C_2\{\Delta_2/y\} \equiv \text{sum}(s(x), y, s(z)) \leftarrow \text{sum}(x, y, z) \\ C_3\{a(C_3)\} &= C_3\{\Delta_3/x\} \equiv \text{sum}(s(x), y, s(z)) \leftarrow \text{sum}(x, y, z) \end{aligned}$$

Therefore

$$P_a\Theta = \left\{ \begin{array}{l} \text{sum}(0, x, x) \leftarrow \\ \text{sum}(s(x), y, s(z)) \leftarrow \text{sum}(x, y, z) \end{array} \right\}$$

The composed LOS also represent an operational characterization of the subsumption relation among programs. The main results are theorems 4 and 5.

Theorem 4. Let P_1 and P_2 be two programs and Θ a composed LOS. If $P_2\theta \subseteq P_1$ then $P_1 \succeq P_2$, where Θ is applied to P via an appropriate amfa.

The following corollary is immediate.

Corollary 1. Let P_1 and P_2 be two programs and $\Theta_1 \dots \Theta_n$ a finite chain of composed LOS. If $P_2\Theta_1 \dots \Theta_n \subseteq P_1$ then $P_1 \succeq P_2$, where each Θ is applied via an appropriate amfa a_i .

The next result is the converse of the corollary 1.

² We use the Prolog notation $A \leftarrow B_1, \dots, B_n$ instead of $\{A, \neg B_1, \dots, \neg B_n\}$

Theorem 5. *Let P_1 and P_2 be two programs. If $P_1 \succeq P_2$ then there exists a finite chain of composed LOS $\Theta_1 \dots \Theta_n$ such that $P_2\Theta_1 \dots \Theta_n \subseteq P_1$, where each Θ is applied via an appropriate amfa a_i .*

The proof of this theorem can be obtained straightforwardly from the theorem 1. If $P_1 \succeq P_2$ then for each $C \in P_2$ there exists $D \in P_1$ such that $D \succeq C$ and we can find a finite chain of LOS $\{\Delta_1/x_1\}, \dots, \{\Delta_n/x_n\}$ such that $D = C\{\Delta_1/x_1\}, \dots, \{\Delta_n/x_n\}$. By joining appropriately the LOS from these chains we have the composed LOS.

6 Quantifying Closeness among Programs

If P_1 subsumes P_2 we can find a finite chain of composed LOS which maps P_2 onto a subset of P_1 . This chain has not to be unique. In a similar way to the clausal case, the shortest chain quantifies the closeness between programs. We formalize this idea in the next definitions.

Definition 9. *Let P_1 and P_2 be two programs such that $P_1 \succeq P_2$. We will say that $\mathcal{C} = \langle \langle \Theta_1, a_1 \rangle, \dots, \langle \Theta_n, a_n \rangle \rangle$ is a chain from P_1 to P_2 if*

- $\Theta_1, \dots, \Theta_n$ are composed LOS.
- For all $i \in \{1, \dots, n\}$, $a_i : P_2\Theta_1 \dots \Theta_{i-1} \rightarrow \Theta_i$ is an amfa.
- $P_2\Theta_1 \dots \Theta_n \subseteq P_1$ where the composed LOS have been applied via the correspondent a_i .

In this case we will say that $\mathcal{C}$ is a chain of length n and we will denote it by $|\mathcal{C}| = n$. If $P_1 \subseteq P_2$ we will say that the empty chain, of length zero, is a chain from P_1 to P_2 . The set of chains from P_1 to P_2 will be denoted by $\mathbf{L}(P_1, P_2)$.

If $P_1 \succeq P_2$, the set $\mathbf{L}(P_1, P_2)$ is not empty and the next definition makes sense.

Definition 10. *We will define the mapping $dp : \mathbb{P} \times \mathbb{P} \rightarrow [0, +\infty]$ as follows:*

$$dp(P_1, P_2) = \begin{cases} \min\{|\mathcal{C}| : \mathcal{C} \in \mathbf{L}(P_1, P_2)\} & \text{if } P_1 \succeq P_2 \\ +\infty & \text{otherwise} \end{cases}$$

The mapping dp verifies the following properties:

- $P_1 \subseteq P_2 \Leftrightarrow dp(P_2, P_1) = 0$, in particular, $dp(P, P) = 0$
- In general, $dp(P_1, P_2) \neq dp(P_2, P_1)$, $P_1, P_2 \in \mathbb{P}$
- $dp(P_1, P_2) \leq dp(P_1, P_0) + dp(P_0, P_2)$ for all $P_1, P_2, P_0 \in \mathbb{P}$

hence, dp is a *pseudo-quasi-metric* and $(\mathbb{P}, dp)$ is a *quantitative domain*.

The next equivalence summarize our study about the generalization of programs under subsumption, by putting together our operators, the subsumption relation, the semantics of definite programs and the weak metric dp .

Theorem 6. *Let P_1 and P_2 be two definite programs. The following sentences are equivalent:*

- For all interpretation I , $T_{P_2}(I) \subseteq T_{P_1}(I)$.
- $P_1 \succeq P_2$.
- There exists a finite chain of composed LOS $\Theta_1 \dots \Theta_n$ such that $P_2\Theta_1 \dots \Theta_n \subseteq P_1$, where each Θ is applied via an appropriate amfa.
- $dp(P_1, P_2) < +\infty$.

Since the programs are finite set of clauses, the next theorem provides a method to compute the pseudo-quasi-distance of two programs.

Theorem 7. *Let P_1 and P_2 be two programs.*

$$dp(P_1, P_2) = \max_{D \in P_2} \left\{ \min_{C \in P_1} \{dc(C, D)\} \right\}$$

Note that the mapping $dp^*(P_1, P_2) = \max\{dp(P_1, P_2), dp(P_2, P_1)\}$ is the Hausdorff metric between programs based on the quasi-metric dc .

7 Related Work and Examples

The problem of quantifying the closeness among clauses has already been studied previously by offering distinct alternatives of solution to the problem. In the literature, a metric is firstly defined on the set of literals and then, the Hausdorff metric is used to get, from this metric, a metric on the set of clauses.

In [8], Nienhuys-Cheng defines a distance for ground atoms

- $d_{nc,g}(e, e) = 0$
- $p/n \neq q/m \Rightarrow d_{nc,g}(p(s_1, \dots, s_n), q(t_1, \dots, t_m)) = 1$
- $d_{nc,g}(p(s_1, \dots, s_n), p(t_1, \dots, t_n)) = \frac{1}{2n} \sum_{i=1}^n d_{nc,g}(s_i, t_i)$

then she uses the Hausdorff metric to define a metric on sets of ground atoms

$$d_h(A, B) = \max \left\{ \max_{a \in A} \{ \min_{b \in B} \{ d_{nc,g}(a, b) \} \}, \max_{b \in B} \{ \min_{a \in A} \{ d_{nc,g}(a, b) \} \} \right\}$$

The aim of this distance was to define a distance between Herbrand interpretations, so $d_{nc,g}$ was only defined on ground atoms. In [10], Ramon and Bruynooghe extended it to handle non-ground expressions:

- $d_{nc}(e_1, e_2) = d_{nc,g}(e_1, e_2)$ if e_1, e_2 are ground expressions
- $d_{nc}(p(s_1, \dots, s_n), X) = d_{nc}(X, p(s_1, \dots, s_n)) = 1$ with X a variable.
- $d_{nc}(X, Y) = 1$ and $d_{nc}(X, X) = 0$ for all $X \neq Y$ with X and Y variables.

This metric can be easily extended to literals: If A and B are atoms, we consider $d_{nc}(\neg A, B) = d_{nc}(A, \neg B) = 1$ and $d_{nc}(\neg A, \neg B) = d_{nc}(A, B)$. By applying the Hausdorff metric to d_{nc} we have a metric d_h on clauses. We have implemented dc and d_h with Prolog programs. The following example allows us to compare this metric with our quasi-metric.

Table 1. Comparison of dc vs. d_h

N	$dc(C_n, D_n)$		$d_h(C_n, D_n)$	
	Sec	Q-dist	Sec	Dist
64	0.02	3	0.11	$\sim 2.7 \cdot 10^{-20}$
128	0.06	3	0.21	$\sim 1.4 \cdot 10^{-39}$
256	0.1	3	0.43	$\sim 4.3 \cdot 10^{-78}$
512	0.26	3	0.93	$\sim 3.7 \cdot 10^{-155}$
1024	0.67	3	2.03	$\sim 2.7 \cdot 10^{-309}$

For all $n \geq 0$, consider the clauses

$$\begin{aligned} C_n &\equiv \text{sum}(s^{n+1}(x_1), s^n(y_1), s^{n+1}(z_1)) \leftarrow \text{sum}(s^n(x_1), s^n(y_1), s^n(z_1)) \\ D_n &\equiv \text{sum}(s^{2n+1}(x_2), s^{2n}(y_2), s^{2n+1}(z_2)) \leftarrow \text{sum}(s^{2n}(x_2), s^{2n}(y_2), s^{2n}(z_2)) \end{aligned}$$

and the substitution $\theta_n = \{x_1/s^n(x_2), y_1/s^n(y_2), x_3/s^n(y_3)\}$. Then $C_n\theta_n = D_n$ for all n and hence, $C_n \succeq D_n$. Table 1 shows the values of the quasi-metric $dc(C_n, D_n)$ and the metric $d_h(C_n, D_n)$ for several values of N as well as the time of computation on a PIII 800 Mhz. in an implementation for SWI-Prolog 4.0.11. It can be easily calculated that, for all $n \geq 0$, $dc(C_n, D_n) = 3$. If we use the Hausdorff metric d_h based on d_{nc} we have that, for all $n \geq 0$, $d_h(C_n, D_n) = \frac{1}{2^{n+1}}$ which tends to zero in spite of the subsumption relation holds for all n .

In the literature, other formalizations of the closeness among clauses (e.g. [4] or [10]) can be found.

If we consider now the clauses

$$C'_n \equiv \text{sum}(0, s^n(u_1), s^n(u_1)) \quad \text{and} \quad D'_n \equiv \text{sum}(0, s^{2n}(u_2), s^{2n}(u_2))$$

and the programs $P_n^1 = \{C_n, C'_n\}$ and $P_n^2 = \{D_n, D'_n\}$ we have that, for all $n \geq 0$, $dp(P_n^1, P_n^2) = 3$ and

$$d_{hh}(P_n^1, P_n^2) = \frac{1}{2^{n+1}}$$

where d_{hh} is the Hausdorff metric associated to d_h .

8 Conclusions and Future Work

The operators presented in this paper might provide a general framework to specify learning process based on Logic Programming [3]. The operators are not related to any specific system, they can be easily implemented and used in any system. But the main property is that the LOS are sufficient for all generalization process of clauses based on subsumption. As we have showed, the LOS are a *complete* set of generalization operators.

We define a quasi-metric on the set of clauses and give an algorithm to compute it as well as a method for a quick estimation. The process of quantifying qualitative relations (as subsumption) is a hard and exciting problem which

arises in many fields of Computer Science (see [6]) which is far from a complete solution. We present a contribution to its study by defining a quasi-metric on the set of clauses in a natural way, as the minimum number of operators which map a clause into another. As we have seen, this quasi-metric considers the clauses as members of a net of relations via subsumption and overcomes the drawbacks found in others formalizations of closeness.

The definition of quasi-metric is completed with an algorithm to compute it and a bound for a quick estimation. This estimation can be a useful tool for the design of new learning algorithms based on subsumption.

In the second part of the paper we present a family of operators which also represents an operational characterization of the subsumption between programs. These operators provide a weak metric which captures the idea of closeness among programs based on subsumption. The main results about generalization of programs are summarized in the theorem 6. The relation between the composed LOS and T_P opens a door for studying in the future new links between these operators and the semantics of logic programs.

References

1. M.R.Garey and D.S. Johnson: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, New York, 1979.
2. M.A. Gutiérrez-Naranjo, J.A. Alonso-Jiménez and J. Borrego-Díaz: A topological study of the upward refinement operators in ILP. In ILP 2000, Work in progress track.
3. M.A. Gutiérrez Naranjo. Operadores de generalización para el aprendizaje clausal. Ph.D. Thesis. Dept. of Computer Science and AI. University of Seville, 2002.
4. A. Hutchinson: Metrics on Terms and Clauses. ECML-97, LNCS 1224, Springer, 1997.
5. P.D. Laird: Learning from Good and Bad Data. Kluwer Academic Publishers, 1988
6. R. Lowen: Approach Spaces, the Missing Link in the Topology-Uniformity-Metric Triad. Oxford Mathematical Monographs, Oxford University Press, 1997.
7. M.J. Maher: Equivalences of Logic Programs. In Foundations of Deductive Databases and Logic Programming. J. Minker ed, Morgan Kaufmann, 1988
8. S-H. Nienhuys-Cheng: Distance between Herbrand interpretations: a measure for approximations to a target concept. Technical Report EUR-FEW-CS-97-05. Department of Computer Science, Erasmus University, the Netherlands, 1997.
9. G.D. Plotkin: A Note on Inductive Generalization. In Machine Intelligence 5, pp.: 153-163. Edinburgh University Press, Edinburgh, 1970.
10. J. Ramon and M. Bruynooghe: A framework for defining distances between first-order logic-objects. Report CW 263, Dept. of Computer Science, KU Leuven, 1998.
11. E.Y. Shapiro: Inductive Inference of Theories from Facts. Technical Report 624, Department of Computer Science, Yale University, New Haven, CT, 1981
12. M. Schmidt-Schauss. Implication of clauses is undecidable. Theoretical Computer Science, 59-3, pp. 287-296, 1988.
13. P.R.J. van der Laag, S.-H. Nienhuys-Cheng: Completeness and properness of refinement operators in Inductive Logic Programming. Journal of Logic Programming, Vol 34, n.3, pp.: 201-225, 1998.

Modeling with Archetypes: An Effective Approach to Dealing with Complexity

Markus Schwaninger

Institute of Management, University of St. Gallen,
Dufourstrasse 48, CH-9000 St. Gallen, Switzerland
Markus.Schwaninger@unisg.ch

Abstract. In the face of growing complexities, agents who lead or manage organizations must revert to better models. This proposition is based on the Conant/Ashby Theorem, which says that the results of a management process are determined by the quality of the model on which that process is based. The author proposes that archetype-based modeling is a promising way to enhance the behavioral repertory of agents in organizations and society: It aids in achieving better models and thus in coping with complexity more effectively. Experiences with System Dynamics modeling of real-world issues and problems as well as the pertinent results achieved are reported. The special flavor of the cases described lies in a new quality and speed of learning-by-modeling, which was unachievable a few years ago. This is now enabled by a) an advanced methodology of modeling and model validation, b) the conceptual maturity of systems archetypes, and c) the availability of powerful simulation software.

1 Introduction: Objectives and Motivation

The speed and uncertainty of events have grown, and with them the need for a better understanding of dynamic complexity as a basis for managerial action. Therefore, models of complex systems which link qualitative reasoning and quantitative decision support have become increasingly necessary in the domain of management. However, areas such as strategic management and organization design have been dominated by verbal argumentation. These fields have shown resistance to quantitative modeling. One of the reasons is that many researchers and practitioners have considered the path toward solid, quantitative decision support too difficult or even impossible in the face of organizational complexity.

The purpose of this paper is to show that "systemic archetypes", as proposed by Senge [14] and Wolstenholme [18], can open a new way to deal with such complexity effectively and efficiently.

This paper emanates from the tradition of System Dynamics modeling and simulation, a powerful stream of the Systems Approach [cf. 13]. In section 2, an overview of modeling and simulation with System Dynamics will be presented. The theoretical basis of the postulate for better models, which has motivated this paper, is the *Conant-Ashby-Theorem*. This theorem will be expounded on – in connection with the concepts of *essential variables* and *archetypes* – in section 3. An account of the

applications of archetype-based modeling and simulation – the core of this paper – will be provided in section 4. The paper is finalized with a discussion in section 5 and the conclusions in section 6.

2 System Dynamics Modeling and Simulation

System Theory has bred a very powerful approach for enabling a better understanding of dynamic complexity in organizations and society – System Dynamics (SD). With "SD", the author refers to a modeling and simulation methodology which originated from M.I.T.-Massachusetts Institute of Technology, where it was invented by Professor Jay W. Forrester [6,7,8].

In SD, systems are modeled as closed loop systems, which largely generate their dynamics internally. They are essentially made up of stocks and flows (→ differential equations); temporal delays are carefully modeled. Given its generic features and the strong software tools available (Stella/Ithink, VENSIM, Powersim, MyStrategy, Jitia), the SD methodology is extraordinarily efficient at grasping the dynamics of complex systems and exhibits exceptional strengths for modeling and policy analysis with regard to socio-technical systems. Applications in organizations have been subject to sustained growth. Factors which appear to restrict the diffusion of SD are a somewhat limited availability of conceptual knowledge leading to good models and a certain resistance on the part of practitioners to engage in comprehensive, methodologically rigorous "modeling exercises".

Recent developments in literature, software technology, and the worldwide exchange of knowledge across the growing community of system dynamicists will gradually eliminate these restrictions. In addition to the journal of that community, the System Dynamics Review, the cumulated knowledge on System Dynamics modeling and simulation is made available in excellent introductions [e.g. 12, 10], text and handbooks [e.g. 15, 11, 16], as well as by means of a continuously updated bibliography edited by the International System Dynamics Society [17]. Standard software packages – Stella/Ithink, Vensim, PowerSim, MyStrategy and Jitia – facilitate the access to the modeling and simulation technology. Generally, the software enables the design of user-friendly interfaces ("cockpits"). The packages also provide features with varying emphases, enabling careful model validation. Certain packages include modules with standard models for specific purposes, optimization heuristics, etc.

3 Modes of Coping with Complexity

Why is the quality of models so important? The answer is given by the Conant/Ashby Theorem from Cybernetics. It says: "Every good regulator of a system must be a model of that system." [5] In other words, the results of a management process cannot be better than the model on which it is based, except by chance. In their work, Conant and Ashby provide the proof of the claim that model quality is crucial. Consequently, the Conant-Ashby-Theorem is as important for management as are the Laws of Thermodynamics are for Mechanical Engineering.

Yet, the use of trivial models in organizational practice still abounds. Models tend to be static; dynamic complexity is not adequately taken into account. Arguments tend to lack clarity; they are not sufficiently underpinned by explicit models. Vagueness often dominates strategy discussions. Hypotheses, if there are any, are not properly tested.

When confronting complex issues, the discussions tend to mire down in two classical traps. The first trap manifests itself in the shape of a symptom *sweeping arguments*, such as: “This issue is highly complex. Therefore, let us simply assume that ...” This is the trap of *trivialization*. The second trap is accompanied by a syndrome, – the illusion of “mastering complexity”, expressed in language figures such as: “To capture the complexity of the real system, we need complex models”, meaning large numbers of variables and formulas. This is the trap of *model complexity*.

Risk Management Theory has developed a rigorous approach for dealing with the risks of model complexity, expressed in the construct of *model risk*. System Theory offers two major strategies for dealing with the challenge of modeling complex systems adequately. One is Ross Ashby’s notion of *essential variables*. Essential variables are those “which are closely related to the survival [of the system under study] and which are closely linked dynamically so that marked changes in any one leads sooner or later to marked changes in the others.” [1: 42]

The other is the concept of *Systemic Archetypes*. The term comes from the Greek *arché* – very old, original, and *typos* – type. Archetypes in the sense used here are ideal-typical (in the sense of the sociologist Max Weber’s *Idealtypen*) patterns which can be repeatedly discerned in the structures of organizations. These archetypes are a kind of pattern language, similar to those used in architecture and software engineering. In all these cases they are conceptual tools, heuristic devices for coping with real-life complexities: for the sake of both, understanding a complex, dynamical system or situation under study and, if necessary, changing it by means of a systemic design. The SD view emphasizes that events are mere instances of larger patterns of behavior, which on their part, are brought forth by underlying structures.

Both of these strategies are connected, in that a) proper use of archetypes is always grounded in understanding essential variables and their interrelationships, and probably b) essential variables are to a great extent of an archetypal nature. Validation, the process by which confidence is built into models, is at the heart of both modeling approaches, essential variables and archetypes.

The emphasis of this paper is on archetypes, a complete list of which does not exist. However, Wolstenholme has come up with the most complete and generic typology of archetypes to date [18], embracing more specific archetypes, such as those expounded by Senge [14] at an earlier stage. An overview of Wolstenholme’s four generic archetypes, as well as their semi-specific subclasses and specific cases are expounded in Table 1.

A generic solution archetype exists for each generic problem archetype; a detailed treatment is beyond the scope of this paper. The next two sections contain an account on how some of these archetypes have been used and which lessons were learned.

Table 1. Summary of systemic archetypes after Wolstenholme [18]

Generic Problem Archetype	Semi-Generic Archetype	Specific Case
“Underachievement” Archetype	Limits to success, Tragedy of the Commons	Growth and Underinvestment
“Out-of-Control” Archetype	Fixes that Fail, Shifting the Burden, Accidental Adversaries	Criminal Justice, Problem Child Walmart vs. P & G ¹
“Relative Achievement” Archetype	Success to the Successful	VHS vs. Betamax
“Relative Control” Archetype	Escalation, Drifting Goals	Arms Race, Quality Improvement

4 Applications of Archetype-Based Modeling and Simulation

The author recently pursued a path of using System Dynamics, which led to promising results. Together with MBA (Master of Business Administration) students, SD has been used in project seminars to tackle relatively complex issues faced by real-world organizations. These seminars have been held at two Universities, one being the University of St. Gallen, Switzerland, where the students were newcomers to complex systems modeling but had some practical business experience in business or the public sector. The other is an MBA program at the Universidad de los Andes, Bogotá, Colombia, with participants who already had substantial experience in management or engineering.

As the students in these seminars had not been exposed to SD in earlier courses, the following approach was adopted. The objective of the course was that participants a) should enhance their skills in order to gain insight into the structures underlying typical patterns of behavior exposed by organizations and to model them properly, b) they should reach a better understanding of the dynamics of organizations as well as the consequences of interventions, and c) they should get a grasp of the potential as well as the limitations of SD as a tool for the design of solutions. Early on, the students studied Senge’s “archetypal” system structures, which stimulated them to a new perception of issues encountered in the real world. Using this, they chose their own cases, learned how to model, validate models, proceed with policy analysis etc., and finally came up with proposals for the improvement of the systems under study.

To date, four seminars of this kind have been held, three at the University of St. Gallen and one at the Universidad de los Andes. Altogether, 13 cases were studied. Three of them proved to be so rich that the work groups have continued to study them in a second seminar, during the following semester. This seminar is still continuing in the 2003 summer term. In the next section, one of the cases studied will be expounded in sufficient detail, and with the dynamics it exhibits, as well as the policy analyses and recommendations made. The rest of the cases will be summarized.

¹ P&G stands for Procter and Gamble.

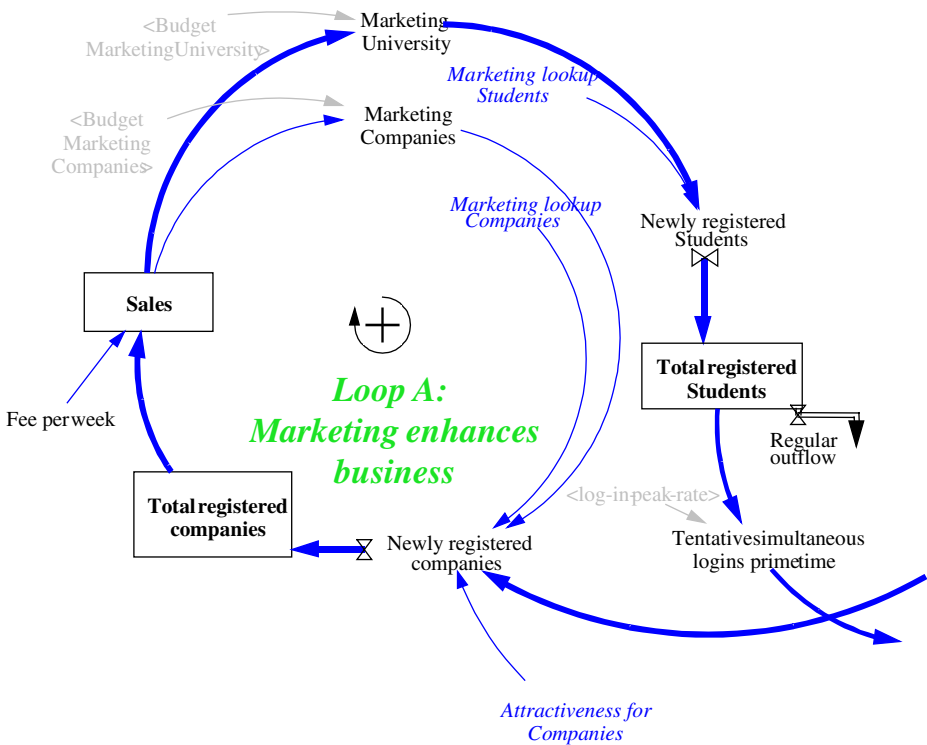


Fig. 1. Loop A – Marketing enhances business

4.1 Exemplary Case: Growth and Underinvestment

The firm studied was a young company – a job agency for university students, an electronic business in Germany. The purpose of the project was to understand the problem of the hampered evolution of the business manifest in that small firm, and to improve resource allocation. Early on the project team identified an instance of growth and underinvestment [cf. 14: 389f.].

The three constituent parts of the stocks-and-flows diagram for the model (elaborated by means of the VENSIM software) illustrates the problem which occurred – a pattern of growth phases alternating with set-backs. The structure is made up of three loops, one of them of the self-reinforcing type (Figure 1), and two of the balancing (self-attenuating) type (Figures 2 and 3). On the one hand, the job agency boosted growth by marketing its services to both, students interested in finding jobs and companies who would offer positions (Figure 1).

On the other hand, the growth was hampered by bottlenecks in the server capacity, which eventually led to frustration and departure by some of the users (Figure 2).

The deeper cause can be found in the third loop, which unveils a problem of lagging investments leading to temporal capacity shortages (Figure 3). The need to invest is perceived too late, and availability of additional servers only materializes after a time lag. As a consequence, the ensuing growth of capacity is poorly

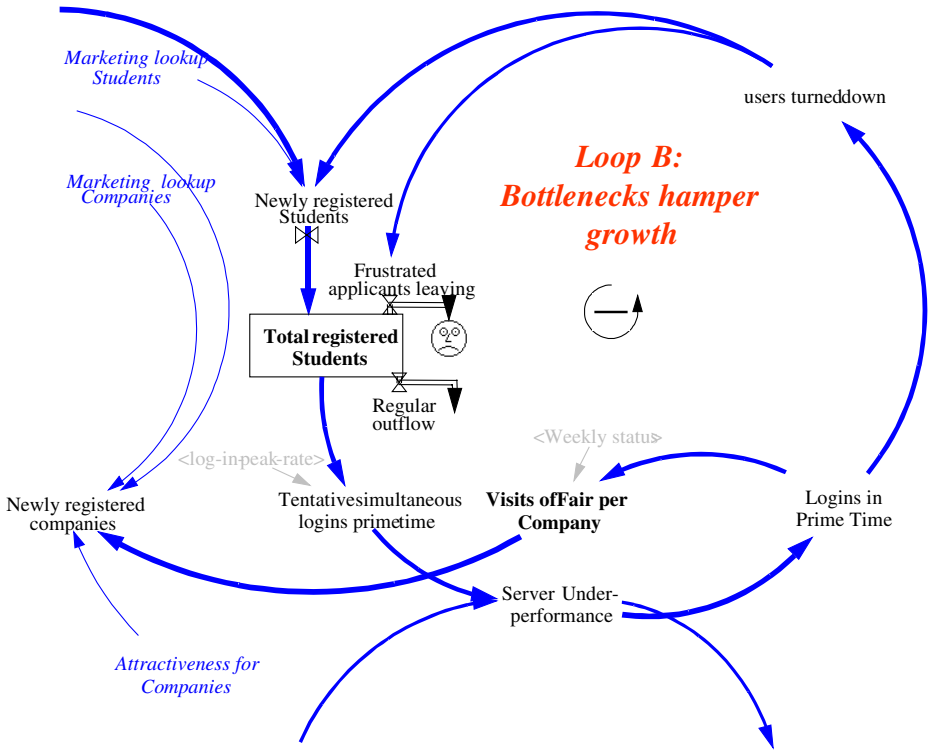


Fig. 2. Loop B - Bottlenecks hamper growth

synchronized with the growing demand. This is the cause of intermittent phases of server underperformance, with the consequences already indicated: Applicants left unattended leave the loop after some time, – normally after a week.

The model was thoroughly validated by means of standard tests for structural and behavior validity, i.e. the test if the model replicated the behavior of the real system under study [cf. 3].

Figure 4 summarizes the *diagnosis* realized by the project team. It shows the evolution of the total number of registered client companies. The bold curve exhibits the actual numbers over time (bold line), with the kinks representing the setbacks. The punctuated line is a realistic projection into the future on the actual numbers of client companies registered. The attenuation of this S-curve produces a kink in the trend due to the temporal server underperformance. The faint line shows a projection of the potential evolution of the business if these technical barriers to growth did not exist. The shapes of the analogous curves for the numbers of registered students exhibit the same pattern.

After this diagnosis, a number of *scenarios* were explored. Different policies for the elimination of growth barriers were tested. These indicated two directions for improved performance – a more prospective or at least more adaptive investment policy and the deployment of resources on smaller servers. The Investment Policy parameter is a measure of the level of capacity utilization (in terms of the percentage/100), which a new server would reach immediately given the current level

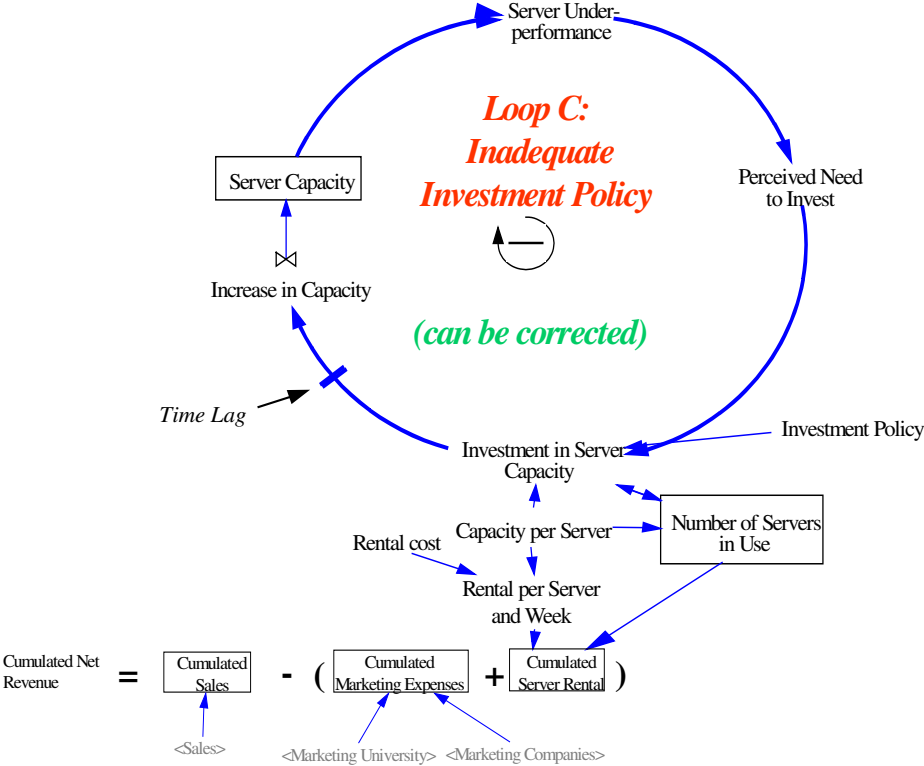


Fig. 3. Loop C – Inadequate Investment Policy

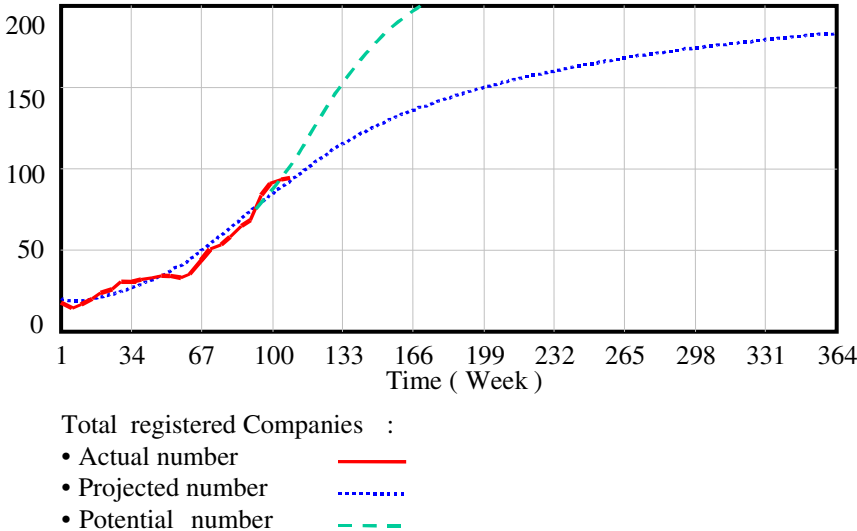
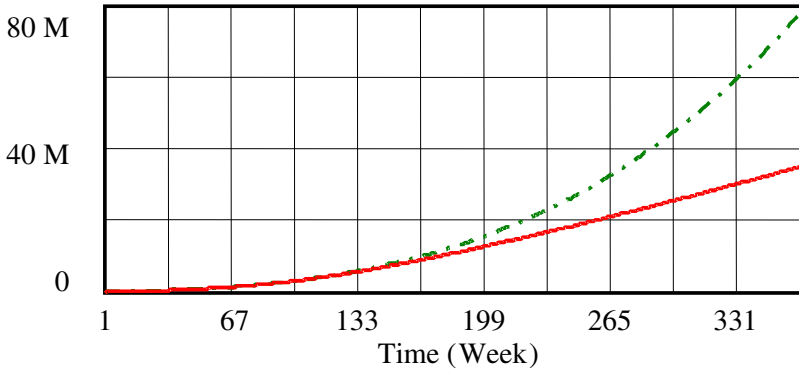


Fig. 4. Number of Registered Companies



Cumulated Net Revenue in Thousand (M) Euro:

Status Quo (Investment Policy=1.0, Capacity per Server=200) ———

Optimum (Investment Policy =0.45, Capacity per Server=130) - - - -

Fig. 5. Investment Strategy - Status Quo versus Optimum

of demand. The Capacity per Server is defined as the number of students that can simultaneously log into a server.

The final *policy recommendation* was elaborated by means of a multiple parameter optimization heuristic, which the VENSIM software makes available on the basis of the Powell algorithm. The parameters jointly optimized were Investment Policy and Capacity per Server. The final policy recommendations are synthesized in Figure 5.

The *optimal strategy* is defined by a more proactive Investment Policy (parameter set at 0.45 instead of 1.0) and the use of smaller servers (capacity for 130 simultaneous users instead of 200). As the rental cost remains invariable for smaller servers, this strategy is counterintuitive: students and participants of executive seminars to whom the model has been demonstrated mostly opted for larger servers in their first guesses.

According to the comments obtained from the project group's internal partners in the firm under study, the model and the policy recommendations assisted the management enormously in gaining an in-depth understanding of the problems they were wrestling with. The model proved very useful as a vehicle for discussions among the people involved.² For that purpose, the process of developing the model, elaborating and validating its structure, as well as exploring its dynamics playfully, was more important than the detailed simulation outcomes were. Several durable insights were gained. Mutual interdependencies, e.g. between internal and external variables, the factors driving and hampering the development path of the company, and the levers of change which could be influenced, became clearer. The building of and the interaction with the model triggered a thinking and communication process which enabled team learning, a process for ultimately and more robustly weathering

² On the part of the firm two managing directors, the CIO (Chief Information Officer) and two further members of the Information Technology department were involved in the project. Altogether, the company has roughly 25 associates, i.e. besides our students about 20% of the staff were involved in the project.

the challenges faced. Today, 15 months after the conclusion of the project, the firm under study appears to have gained in strength and found its place in a highly competitive market. As a supporting factor, this model was one, albeit not the only contribution.

4.2 Further Cases

It is worthwhile enumerating the other cases which were subject to modeling and simulation exercises in the three seminars mentioned. These were³:

Case 2: Tax Competition between Swiss Cantons	I
Case 3: Lake Sempach – Farming and Fishing (Ecological Problem)	I
Case 4: Arms Race during the Cold War	I
Case 5: Dental Clinic – Flow of Patients	II
Case 6: Reorganizing Sales in a Software Distribution Company	II
Case 7: The Problem of Cheating – A Quality Problem in a University	II
Case 8: Marketing Initiatives in a Decentral Organization	II
Case 9: Information Technology – Release Policy in St. Gallen State Government	II
Case 10: Consulting Services during Recession – The Human Resources Issue	III
Case 11: Micro-Mobility: Overshoot and Collapse of a Microscooter Producer	III/IV
Case 12: Pension Funds: Financing and Risk Issues	III/IV
Case 13: Pension Funds: Financing and Risk Issues II	III/IV

Not all of these models achieved the same level in terms of validity, relevance, and usefulness, but all of them improved a great deal as the projects advanced. Several of these cases led to models useful enough to support and probably improve real-life decision-making in the real systems under study. This was the case at least in cases 6 (a problem of the Limits-to-growth type) and 9 (a situation of the Fixes-that-fail-type). Others at least led to insightful improvements of mental models-in-use (Cases 10 and 11). The last three cases were so complex and the students so highly motivated, that the author offered to accompany them for a second seminar in the ensuing term. This seminar is close to completion, as this paper is written. The models have been greatly improved and submitted to thorough validation procedures, which in one case included a readjustment of model boundaries.

5 Discussion

The results achieved with archetype-based modeling in the experiences outlined have been a most positive surprise. The quality of the modeling and simulation work done

³ Cases marked with ,I‘ are from the St. Gallen seminar in Winter 2001/2, with ,II‘ from the Bogotá seminar in Summer 2002, with ,III‘ from the St.Gallen seminar in Winter 2002/3, and with ,IV‘ from the St. Gallen seminar in Summer 2003.

and the strength of the policy recommendations elaborated were astonishingly high, given the short duration of the courses and the limited resources available. The speed of learning of the teams was far superior to cases of comparable traditional courses.

This feature was particularly salient in the Seminar with Colombia, which had to be carried out in a "virtual" mode, using of the new StudyNet platform of the University of St. Gallen. The following communication channels were used: 1. & 2. Audio and video, 3. Application sharing, 4. Chat function, 5. E-mail, 6. Telephone. Despite all the restrictions imposed by these mediated forms of interaction, the students came up with remarkable models, and their recommendations even had some impact on the organizations under study (a University, a software company, a dental clinic etc.).

How was it possible to achieve these results? First of all, the students generally showed excellent motivation and commitment. Second, the archetypes were used as conceptual stimuli to help the phases of issue identification and diagnosis. To avoid constricting the students' thinking it was emphasized, however, from the start that real-life issues may exhibit features which transcend the bounds of one archetype – for example in cases where a combination of archetypes applies. Consequently, the cases listed in the section above are not always as closely confined to one archetype as was the case in the account of the growth and underinvestment example.

Third, throughout the process, the conceptual and methodological guidance of the students was crucial. At the outset, conceptual input and training in the use of the software were provided. Thereupon the emphasis was on learning-by-doing, which was backed by intensive support. Several learning loops were built into the process. Student groups submitted partial results along a schedule followed rather tightly – causal loop diagrams, stock-and-flow diagrams, models at different stages of accomplishment, validation steps and results, setups and results of scenarios, policy tests etc. These were examined by the author or his assistant, and then discussed with the groups. The doors of the supporters were open for consultation. Hands-on support for the groups was provided whenever necessary. Intensive help was needed, for example, for more sophisticated applications such as the use of the optimization heuristics. In each seminar, the intermediate and final results of all groups were presented and discussed in plenary sessions. A rather extensive documentation of the model, the process of diagnosis, design and validation, as well as the final recommendations was handed in by each group, and made available to the partnering firms after the end of the term.

Altogether, this design of the process led to an extensive learning experience for all subjects involved. Perhaps not all of the members of a group were in a position to master the methodology and the techniques of modeling and simulation at the end. However, the opinion leaders in the groups were, and the rest had obtained a good feeling of what can be achieved by means of System Dynamics modeling and simulation, and where the limits of the chosen approach are.

6 Conclusions

The compelling conclusion of this paper is that the SD methodology supported by powerful software and conveyed by a didactically effective (self-)education – learning-by-modeling (and simulation) – can open new dimensions, not only to

coursework in strategy and organization. As the case study above shows, these advantages are now available also to practitioners of management. The stimulation through the initial study of archetypes proved most useful in this context. The motor of the effectiveness of these ventures, however, was in each case a set of powerful brains triggered by appropriate methodological (plus technical) support and coaching on behalf of experienced instructors.

Moreover, this innovative approach shows a new path to anyone – practitioner or theoretician – who deals with the issue of organizational complexity: dynamic modeling is now within easy reach, and it can be put in practice at stunning levels of quality. Generic archetypes are available [14, 18] for facilitating the qualitative understanding of dynamical systems. The methodology of System Dynamics modeling and simulation has matured – generally (cf. 15), and with respect to the improvement of model quality, i.e. model validation (cf. 9, 3, 4). Finally, software technology has advanced enormously, facilitating model building and simulation. Some packages (Vensim, Powersim) close the gap between simulation and optimization (in the sense of optimization heuristics). State-of-the-art software also contains inbuilt features dedicated to supporting model validation (e.g. consistency checks, Reality Check®).

As in computer science, where software developers caught in the complexity trap are increasingly reverting to the use of generic "patterns" in diagnosis and design, and in architecture, where Christopher Alexander already suggested a "pattern language" a generation ago [2], management scientists can finally reach new horizons through the use of generic "archetypes", in their quest for better models.

Acknowledgements. The author extends his thanks to Messrs. Kristjan Ambroz and Camilo Olaya. They assisted him in teaching the seminars and supported the students in St. Gallen and Bogotá with the utmost care and effectiveness. The author is also grateful to the students for their outstanding commitment and enthusiasm in delivering high quality work. The group members for the model referred in section 4.1. were: Clemens Mueller, Anatol Pante, Philipp Tuertscher, Boris Doebler, Sven Graber. Special thanks go to Dr. Bob Eberlein, Ventana Systems, for valuable software support, and to Kristjan Ambroz, who read the manuscript of this paper, for his helpful comments.

References

1. Ashby, W.R.: Design for a Brain, Second Edition, London: Chapman and Hall (1960).
2. Alexander, C., et al.: A Pattern Language, New York: Oxford University Press (1977).
3. Barlas, Y.: Formal Aspects of Model Validity and Validation in System Dynamics. *System Dynamics Review*, Vol. 12, No. 3, (Fall 1996) 183–210.
4. Barlas, Y. and Carpenter, S.: Philosophical Roots of Model Validity – Two Paradigms. *System Dynamics Review*, Vol. 6, No. 2 (Summer 1990) 148–166.
5. Conant, R. C. and Ashby, W.R.: Every Good Regulator of a System Must Be a Model of that System. In: Conant, R., ed.: *Mechanisms of Intelligence. Ashby's Writings on Cybernetics*, Seaside, Ca.: Intersystems Publications, Seaside (1981), 205–214.
6. Forrester, J.W.: *Industrial Dynamics*, Cambridge, Mass.: MIT Press (1961)..
7. Forrester, J.W.: *Principles of Systems*, Cambridge, Mass.: MIT Press (1968)

8. Forrester, J.W.: Counterintuitive Behavior of Social Systems. *Technology Review*, Vol. 73, No. 3 (January 1971) 52–68.
9. Forrester, J.W. and Senge, P.M.: Tests for Building Confidence in System Dynamics Models, in: Legasto Jr., A.A., Forrester J.W. and Lyneis, J.M., eds.: *System Dynamics*, Amsterdam etc.: North-Holland (1980) 209–228.
10. La Roche, U. and Simon, M.: *Geschäftsprozesse simulieren. Flexibel und zielorientiert führen mit Fließmodellen*, Zürich: Verlag Industrielle Organisation (2000).
11. Richardson, G.P., ed., *Modelling for Management. Simulation in Support of Systems Thinking*, 2 Volumes, Dartmouth: Aldershot (1996).
12. Richmond, B.: *An Introduction to Systems Thinking*, Hanover, NH: High Performance Systems (1992–2003)
13. Schwaninger, M.: The Role of System Dynamics within the Systems Movement, in: Theme ‚System Dynamics‘, edited by Yaman Barlas, in: *Encyclopedia of Life Support Systems (EOLSS)*, UNESCO, Chapter 6.63.2.4., Oxford: EOLSS Publishers, WWW.EOLSS.NET (2003/4, forthcoming).
14. Senge, P.M.: *The Fifth Discipline. The Art and Practice of the Learning Organization*, London: Century Business (1992).
15. Sterman, J.D.: *Business Dynamics. Systems Thinking and Modeling for a Complex World*, Boston, Mass.: Irwin/McGraw-Hill (2000).
16. *System Dynamics in Education Project (SDEP MIT): Road Maps – A Guide to Learning System Dynamics*, Cambridge, Mass.: Massachusetts Institute of Technology, <http://sysdyn.clexchange.org/road-maps/home.html> (1996–2000).
17. *System Dynamics Society*, ed.: *System Dynamics Bibliography*, Albany: System Dynamics Society, 2002 (updated regularly).
18. Wolstenholme, E.F.: Towards the Definition and Use of a Core Set of Archetypal Structures in System Dynamics. *System Dynamics Review*, Vol. 19, No. 1 (Spring 2003) 7–26.

Equal Opportunities Analysis in the University: The Gender Perspective

I.J. Benítez¹, P. Albertos¹, E. Barberá², J.L. Díez¹, and M. Sarrió²

¹Department ISA, Polytechnic University of Valencia,
P.O. Box 22012, 46071 Valencia, Spain
{igbesan,pedro,jldiez}@isa.upv.es

²Institut Universitari d'Estudis de la Dona, Universitat de Valencia
Avenida Blasco Ibáñez 32, 46010, Valencia, Spain
{barberah, Maite.Sarrio}@uv.es

Abstract. The social systems' complexity is a consequence of the human presence. Measurements and evaluations are rather qualitative and, in many cases, heuristic and / or linguistic. Human beings are not equally treated, with a clear discrimination based on age, gender, race or culture, among many other reasons. Women discrimination is the focus of our study in this paper. The university, as a social system, is investigated. The goal is to determine the degree of gender discrimination and to provide the tools to evaluate different actions to improve the equal opportunity (EO) principle in its operation.

1 Introduction

The complexity of social systems is a well-known subject, especially when trying to analyse the human behaviour and reactions taking place in the environment of a hierarchical structure, [10], [14]. The difficulty increases when the objective is not only to analyse a given system, but to change the system's conditions acting as a constraint to get some expected results or outcome of a given variable.

The Equal Opportunity (EO) condition claims for an equal treatment of people to get a job or to develop an activity only based on their suitability to the objective requirements and disregarding any human characteristic not being relevant for that purpose. Usual discrimination is based on age, gender, religion, race or cultural habits. Even more, to carry out the different activities in a complex organization requires the concurs of people with a variety of skills, also enriching the quality of the operation and providing a source of innovation and progress.

Our subject in this contribution is the gender perspective and, in particular, the analysis of possible gender discrimination in public institutions like the university. To eliminate or at least reduce the discriminating conditions, different approaches have been proposed. Positive actions or gender in the mainstreaming [4], to keep in mind the gender perspective at any level of decision, are among the best known policies. In this study, the diversity issue is the baseline and main motivation. People around any activity are diverse and activities' characteristics are also diverse, thus, people in charge of these activities should be also diverse to get the best matching between requirements and performances.

Our study is being carried out in the framework of the European project “*Divers@: Gender and Diversity*”¹. Although the project goals are focused to the analysis of our environment, a transnational part deals with the comparison of different societies. In particular, we are collaborating with a Netherlander’s project led by the University of Maastricht. By this analysis we try to detect non EO situations, show up some hidden barriers even they are not allowed by law, and provide a tool to evaluate different corrective actions to eliminate the gender discrimination.

To be able to transform a social system, under limited options and resources, two initial activities should be done: to capture the current status by means of reliable and quantifiable measurements and indicators, and to establish agreed relationships among variables, some of them suitable to be manipulated, [2], [5].

From the diversity perspective, an initial goal or objective is to define the ideal degree of diversity a given task implies. This is rather controversial, changeable and sometimes influenced by stereotypes, but it will provide a target to analyse the EO condition at a workplace. The next issue is to analyse the environment to detect if there are conditions or constraints limiting the access of women to a given position. These constraints may be local (specific of the institution or community under study), arbitrary (introduced without being consistent with the activity to be developed), legal (quite unusual because there have been clear law advances against the discrimination) or social (the most common and difficult to realize and also to change). The basis of these constraints should be analysed and the experts can propose corrective actions at different levels of decision. Training, awareness and dissemination are fundamental actions to achieve a long term consolidated improvement. As a result, a diversity enhancement through time, resulting in better performances, is expected. The success of this approach relies on the availability of data. Numeric data, such as segregating salaries, production and working time by gender, but also subjective information, such as motivation, satisfaction, and other feelings, only obtained by questionnaires and interviews. In this paper, a methodology is presented which attempts to evaluate gender diversity at workplace, gender EO conditions and the existence of barriers or constraints that could be present at an institution, in this case a university. The analysis of the interaction among some of these variables will pave the way to enhance the EO. A software tool is being developed as a support to make easy the data treatment, to visualize the results and to be used as a safe and costless benchmark to evaluate the effects of prospective actions before being on site implemented.

A model of university as a process is presented, with the variables needed for a detection of the barriers for gender EO. Next, the analysis of diversity is explained, detailing the processes or filters that alter its normal evolution. Finally, an enhancement option based on an expert system is suggested. The capture of the expert’s knowledge will provide the basis to suggest specific EO actions for specific areas, based on the measures and the results from an analysis of gender barriers and values of diversity mismatch. A section for conclusions is also included.

¹ This research is partly funded by the European Social Fund Project ES296 Divers@: Gender and Diversity.

2 Analysis of Institutions

Modelling of social systems, as other complex environments, can be done following different approaches. White box (physical) models [7], obtained by using a mathematical-based theory, are not very useful in this case. Black box (universal functions approximators) models, obtained from experimental data, are preferred [9], but it is known that their capability of generalization is very poor. So, a much more reliable approach is to use experts understanding of the system to discover simpler structures and their relationships [10], [14], using experimental data to help in a fine tuning of parameters of the submodels and their interactions. This will be our approach to analyse the gender EO in the universities.

2.1 The University as a Process

A lot of research exists in EO analysis in institutions, [8], [11]. To build up a model of the university we must first define which variables are needed for gender EO analysis. Let us consider the university as a process, with input variables, some interferences, and the measurable outputs, which result from the processing through time of inputs and interferences. This process is depicted in Fig. 1.

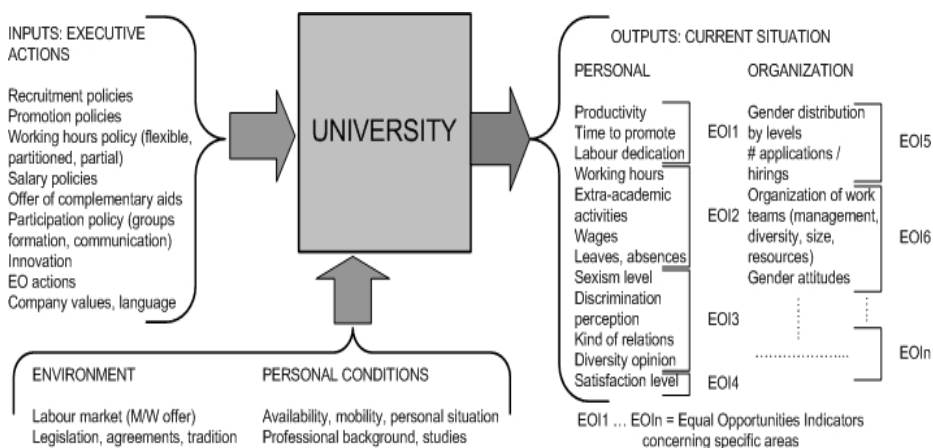


Fig. 1. University I/O model from a gender perspective, with EO indicators

The *inputs* are mainly university policies, all those actions that are applied and can be changed, such as recruitment policies, promotion, salaries, company values regarding gender EO, the presence or absence of specific EO policies, etc.

The *interferences* are facts or events that are given; some of them can be measured, but none of them can be modified at the university level. They are variables representing the individual conditions (such as age, professional background, status, number of children, etc.) or the current market state (labour offer, legislation, agreements).

The *outputs* are variables being measured at a specific period of time, indicating how things are at that moment or how much they have changed, in order to evaluate

the trend. Measurable outputs are collected from the university documents, or by means of questionnaires or discussion groups. The information, therefore, can be of a qualitative or a quantitative nature, and portrays the situation of the workers, such as wages, worked hours, number of absences, productivity or discrimination perception, and the situation of the university itself, such as the gender distribution by levels or departments, or the organization of work teams (gender of the main scientists, group size and gender distribution, resources perceived), which will give a value of women's presence as scientists, [13].

All these variables are regarded as valuable from a gender EO point of view, since the analysis of their current value will allow a diagnosis of gender discrimination through a set of predefined EO indicators (see Sect. 4.1). The accuracy of this diagnosis will strongly depend on the veracity of measured data. Instead of an analysis of these absolute figures, the diversity perspective will allow to better understand the options, to investigate the reasons for possible discriminations, and to act in order to get the best operation of the institution.

3 Diversity Analysis

"Gender in the mainstreaming" is an innovative procedure to deal with gender discrimination [12]. Traditional strategies were 'tinkering' (pursue equal rights and equal treatment, trying to establish formal equality between women and men) and 'tailoring' (also known as positive action and positive discrimination measures, rather than formal equality pursue material equality). 'Transforming the mainstream', rather than seeking to fit women into the systems and structures as they are, pursues a reorganisation of universities in such a way that the demands and expectations of women and men are heard and respected equally.

3.1 The Principle of Diversity

The concept of diversity is a general one applicable to activities, positions and people. Everywhere, but in particular in a large organisation, there is a variety of tasks and attitudes to carry out the required operations. People also have different characteristics, matching these requirements. The diversity principle states that *the best performing institution combines people and tasks, according to their objectives differences and activity*. As the diversity strategy involves the transformation of institutions (eliminating discriminatory elements from existing structures, procedures and customs), their structure has to be analysed.

One institution can be considered as a system composed of interacting subsystems, which can be work teams, departments, collectives of workers, or any other suggested relevant stratification. Of course, different jobs require the use of different tasks and abilities. It would be desirable that all these teams, formed by people (workers), with different backgrounds, personal constraints, expectations, beliefs, formation, etc, develop their different jobs in a balanced *diverse* environment, being all of them plenty satisfied and fully productive. The goal is to match the diversity of abilities and skills needed for different jobs within an institution, with the diversity of mankind or potential workers, applying equal conditions. Although this study focuses on gender

diversity at universities, the evaluation of the results will help to develop a generic tool to analyse diversity within other complex social system.

3.2 The Management of Diversity at Universities

Let us consider a work team, that is, a group of workers that work together or belong to the same level or collective within the university. The workplaces these workers occupy require the use of specific and defined skills. According to the personal abilities required for the job and their psychology stereotyped assignment [6], a desired gender diversity function is generated for the team. This function is a mere Generalized Gauss Bell function with tuneable parameters, being the horizontal axis the proportion of women in percentage (from 0 to 100 %), and the vertical axis a value of adequacy of the current proportion of women to the desired diversity. This adequacy value varies from 0 to 1, meaning 1 if it is totally appropriate, that is, the desirable proportion for optimum diversity. The adequacy value is obtained projecting the current proportion of women at the workplace on the desired diversity function. An example of the desired diversity function and the adequacy value yielded by a current proportion of women at the workplace is depicted in Fig. 2, either using percentage values (**a** axis) or absolute numbers (**b** axis).

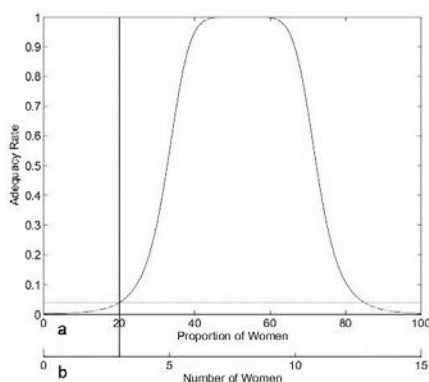


Fig. 2. Adequacy of a given proportion of women on the desired diversity function, in percentage (a) and in absolute numbers (b)

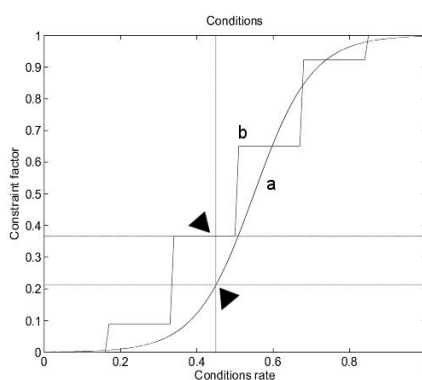


Fig. 3. Conditions rate/constraint factor functions, sigmoid (a) and ladder (b)

In the example depicted in Fig. 2, the desired diversity function has its maximums over the range from a 43% of women up to a 62%, being the centre of the desired rate 52.5%. The current proportion of women is 20% (3 women of a total of 15 workers), with a very low adequacy value of 0.0398 (3.98% of the optimum).

What are the constraints that provoke such a difference between the optimal diversity and the real diversity? The desired diversity has not been implemented due to some constraints. We have identified the *working conditions* a woman may find when applying for a job, and the women's *availability* to meet the requirements for a specific job. The final *selection process* has also been taken into account, since it has

to be considered that diversity can be artificially forced or purposely banned, depending on selection policies and/or prejudices or preferences of the people that select.

Constraints to Diversity. When applying for a job or considering a chance to be promoted, a worker evaluates the advantages and disadvantages the job offers, taking into account personal considerations and the compatibility of the future job requirements with the personal constraints and expectations. In which *conditions* the job is offered are of a great importance, since they will be compared by the applicant with her/his preferences and/or necessities and, if a minimum threshold is not achieved, the potential worker will not accept (nor apply for) the post. Features like the presence or absence of a childcare facility, the flexibility of the working hours or the possibility of home-working can positively or negatively affect the worker's evaluation of the job or even provoke a withdraw.

Mathematically speaking, this conditions rating could be depicted using an incremental function, either continuous or discontinuous. For instance, a sigmoid function (Fig. 3, a) or a ladder function (Fig. 3, b) could be used.

As can be seen in Fig. 3, the horizontal axis represents a numerical rate of the working conditions, rated from 0 (the worst conditions) to 1 (the best conditions). The vertical axis would be an equivalent constraint factor, which rates the preferences or conditions a worker would need for accepting a specific job, and is rated from 0 to 1. In the example from Fig. 3, the conditions are rated with a 0.45, and the constraint factor obtained is 0.2119 (using sigmoid, case a), or 0.2833 (using ladder, case b).

This constraint factor is used to reduce the availability of workers applying for that job. If the constraint factor is near 0 almost no one will apply.

The *offer of the market* has also to be considered, because it is a reality that not always candidate women and men are encountered in the same proportion. Due to some reasons, such as the existence of "traditional" men or women careers, or the fact that men have usually had more chances to develop professionally, it may happen that only very few women have the formation degree or level of studies required for a specific job, so no woman can apply for it. Even having got the required titles, there exist limitations in form of experience, know-how, particular skills or others, that make them not to meet all the requirements for a specific job offer.

Not only the market offer affects the availability: if previously defined constraint factors are high, some workers that could apply for the job will decide not to do it, reducing, thus, the number of available candidates. It may occur than the number of available women is less than the number of women needed according to the optimal diversity, therefore diversity becomes constrained.

The availability has been represented as a decremental function with three parameters: percentage of women that completely fulfils the requirements, maximum number of women that at least partially meets the requirements and curve slope. The range of women available is obtained from the product of the women available (from the study of the market and the offer), and the constraint factor, obtained from the analysis of the conditions, as previously seen. An example of this modified availability function can be observed in Fig. 4, a.

In the example used in Fig. 4, the initial offer of women was 30. Taking as the constraint factor the value shown in Fig. 3, a, yields a final offer of 6.36 women. Conditions modify availability, and modified availability alters desired diversity. As a result, the desired diversity function becomes reshaped into a new function, called

constrained diversity (Fig. 4, b). The constrained diversity does not represent the desired number of women, but the diversity which is currently available, after conditions and availability have been processed. The constrained diversity function is obtained normalizing the product of availability function (Fig. 4, a) and desired diversity function (Fig. 2, b). If availability of women meeting all the requirements exceeds the range of desired diversity, the resulting constrained diversity will be exactly the same as the desired one, meaning that it is possible to obtain the desired diversity. On the other hand, when the number of women available is lower than necessary, desired diversity aspect will be similar to result shown in Fig. 4, b.

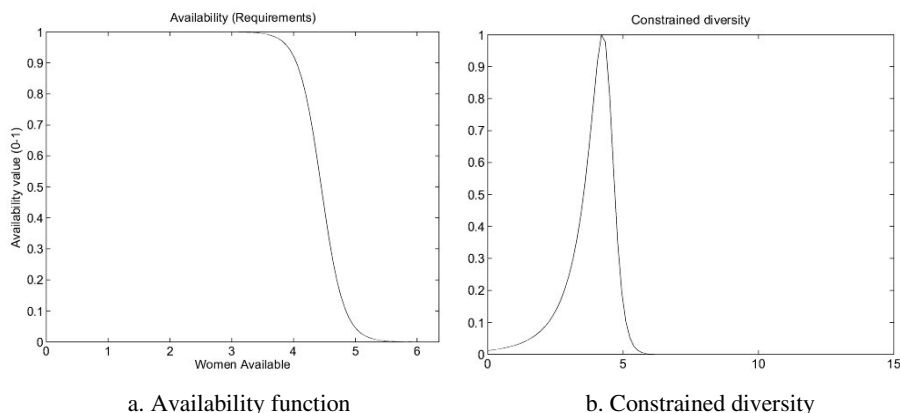


Fig. 4. An example of constrained diversity computation (b) from a desired diversity and the availability limited by offer and conditions rate (a).

Selection Process. After constrained diversity is obtained, a selection process is done. This process will look to settle a work team attending to different principles: to obtain diversity from the available range previously determined, to increase the presence of women by means of positive actions, to increase men presence due to discrimination or simply to alter the expected proportion of women and men due to other reasons out of gender discrimination.

The first option is to form the work team based on constrained diversity. However, a positive action might be included at this point, which would help to increase the final proportion of women in the work team apart from post requirements and other factors. This would result in a function similar to that of the conditions, where the higher the score is over the horizontal axis, the higher the value of the vertical axis results, turning this value into a proportional displacement of the constrained diversity function along its horizontal axis, having thus more availability of women. In the case of a selection process where gender discrimination exists, the number of women would probably increase less than necessary from diversity point of view. Other factors far from gender, such as budgets, race, etc. could additionally affect the selection.

In any case, a selection would be done and a new team would be settled, having a new proportion of men and women. The adequacy value of this proportion will tell how far or how close the new proportion is from the initial desired diversity. The whole process of the management of diversity is depicted in Fig. 5.

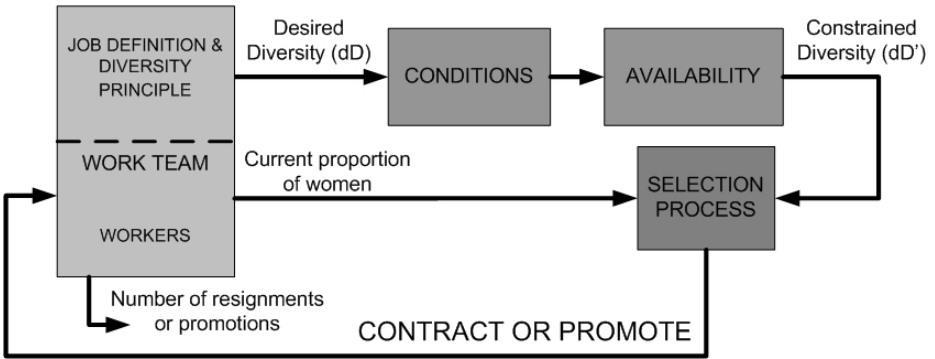


Fig. 5. Management of diversity at a work team

3.3 Example

A comparison between two different conditions rates will show how these affect the process of managing diversity. Taking the example that has been represented at the figures to this point; assuming that the selection process has been made attending strictly to gender equity, and that the three women that were already at the team keep on working, after a constraint factor of 0.21 (Fig. 3, a) and a slight positive action, the resulting constrained diversity would allow only two women to be hired or promoted (see Fig. 6, a), having thus five women at the workteam (33.33% of workers), yielding an adequacy value of 0.55 (Fig. 6, b).

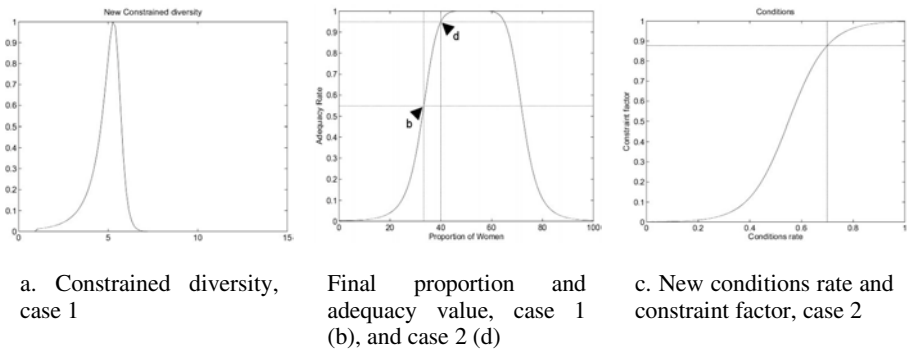


Fig. 6. Comparison of the results from two different conditions rates

Given the case that, for instance, having the same premises, the conditions rate raises up to 0.7, the equivalent constraint factor would be 0.87 (Fig. 6, c). The new availability would be higher, allowing a better selection from a gender EO perspective. As a result, the new team would have six women (40%), yielding an adequacy value of 0.94, which is very close to desired diversity (Fig. 6, d).

4 EO Analysis at the University

Diversity, and in particular the matching between the optimal and the actual diversity, is a global EO measure, but to evaluate the institution opportunities, some other indicators should be taken into account. These indicators will assess the situation and forecast the evolution. For that purpose, in order to evaluate the EO at the university, a number of variables are gathered in groups or areas, according to a established relation concerning gender EO (see groups of variables for EO indicators in Fig. 1), as seen in Sect. 2. The analysis of these groups of variables separately will yield values of EO indicators for each specific area. The value of each indicator is a measure of gender discrimination in that specific area, [4].

For instance, there will be an EO indicator concerning the relation between hours worked and wages perceived, obtained from the analysis of the variables working hours, extra-academic activities, leaves or absences, and wages perceived (Fig. 1, EOI2), segregated by level or position and gender. Another example is an EO indicator of access, obtained from the analysis of the gender distribution by levels, and the number of applications, promotions and hiring, segregated by gender (Fig. 1, EOI5). The access EO indicator will give a measure of how equal/unequal the opportunities for men and women are to apply for a certain position and to promote to upper levels.

Some other classifications would be done, having as a result a bunch of EO indicators, informing of equalities/inequalities concerning gender within every area considered relevant by the experts. The values of these indicators should be obtained comparing gender-segregated data. The method here proposed to obtain the values of the EO indicators is by means of fuzzy clustering techniques.

4.1 The Use of Clustering in EO Analysis

Fuzzy clustering techniques are a powerful tool to detect inequalities [1]. The fuzzy clustering algorithms divide data into clusters, according to proximity or similarity in a mathematical sense, [3].

The example depicted in Fig. 7 shows how a fuzzy clustering algorithm would be applied on a set of segregated data from men and women concerning the level they belong to, and the wages they perceive. The comparison between pairs of resulting clusters will determine how equal or unequal the opportunities are for women to promote to higher levels as men do, and to perceive the same wages as men perceive.

The values of the EO indicators are obtained applying these algorithms [3], to gender segregated data provided by the experts. Each group of data concerning a specific EO indicator is segregated by gender and fuzzy clustered. The number of clusters is either established by the experts or obtained by means of a certain validity criterion, prior to the clustering process. The result will be gender-segregated data grouped into a pre-fixed number of different clusters. The comparison of clusters of women's data with clusters of men's data will show how different or separated they are, and will yield a measure of gender equality/inequality for that specific characteristic or group of characteristics, such as, for instance, salary.

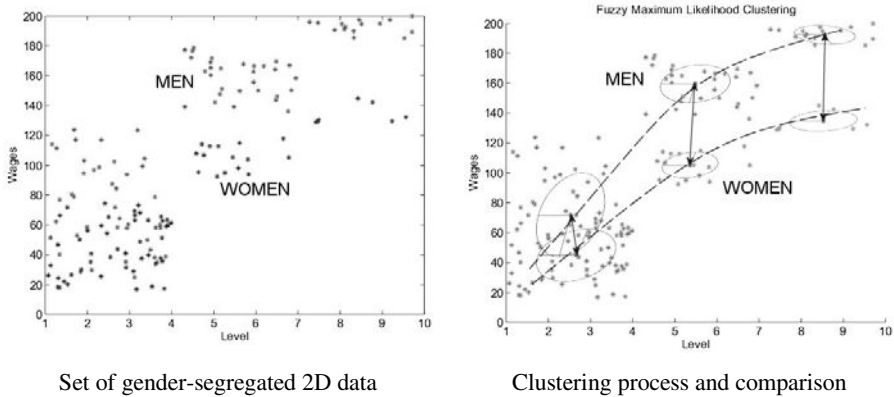


Fig. 7. Fuzzy clustering of a 2D set of data

One by one, the rest of EO indicators would be obtained. A global EO indicator will then be computed from the weighting of all specific EO indicators, such as shown in Formula (1). The different weights, decided by the experts, allow to point out the relevance of a specific aspect to be considered as discriminating.

$$\text{Global EOI} = \sum_{i=1}^n w_i * \text{EOI}_i . \quad (1)$$

In this way, the current situation of the university can be estimated. Also, the influence of the different data in the final result can be evaluated by simulation and some critical issues can be detected.

When all variables and EO indicators are available, clustering techniques can also be applied to determine conditions and/or availability function. For instance, data can be arranged as number of applicants (part of EO15, Fig. 1) versus requirements (EO11) and clustered in order to determine availability function. Conditions function can be identified by clustering a combination of opinion and satisfaction (EO13 and EO14) versus conditions (EO12).

4.2 Barriers Analysis

Using the previous model, the influence of the different data in the final result can be evaluated by simulation and some critical issues can be detected. It is of particular interest to analyse if there are some working condition constraints limiting the access of women to higher positions. The popular concept of “*glass ceiling*”, can be illustrated as far as it could be shown that some apparently neutral requirements to reach a given position result in a step in the constraints that few women accept.

An analysis of the barriers the women met to progress in their working place can be carried out combining different EO indicators, parameters and external conditions.

The analysis of the EO indicators, mainly if a barrier is detected, combined with the degree of mismatching between the optimal and actual diversity, as depicted in Fig. 8, will be the basic information to postulate different corrective actions. It is clear that there is a huge variety of possible actions (local, global, short-term, long term,

individual, social) and also that they strongly depend on the awareness of the social agents. Our proposal is to capture the main operating rules (very sensible to personal and ideological positions) and implement an expert system to help in the decision process.

One advantage of the availability of this model is the possibility to test different scenarios, validating the model and selecting the best actions.

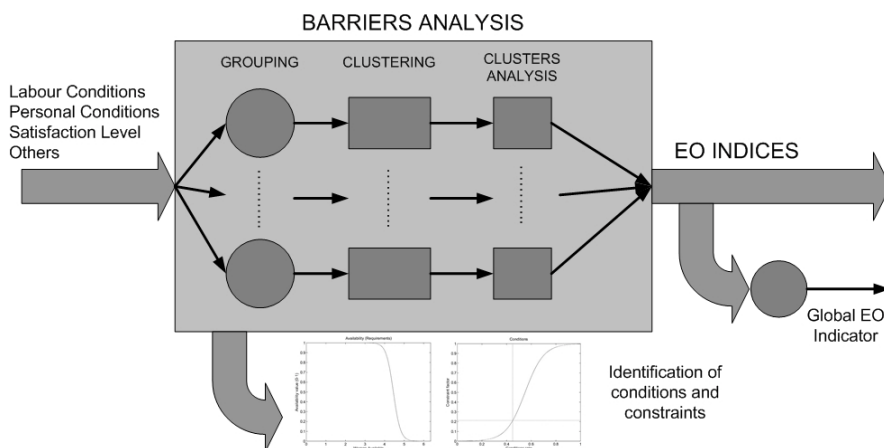


Fig. 8. Barriers analysis

5 Conclusions

A model-based tool to analyse the working conditions and, as a result, the gender EO situation in a social system has been presented. The university has been chosen as the institution to be analysed. Externally, it appears that there is no gender discrimination in any legal or internal operation guidelines, but all we know that there is not a balanced presence of men and women, mainly in the higher decision positions. To quantify and measure the different variables, allows for a formal analysis of the actual situation and to know in which point of the decision chain there is a gap or, in general, a barrier for the women to progress.

It is also evident that there are many subjective parameters and variables in the model, but as far as they are explicated and their influence can be realised, a better understanding of the whole situation is provided and the information to make adequate decisions is available.

Although the concrete example is related to the university (and a practical experiment is under development to analyse the EO at the level of the rector team) the ideas behind this application can be extended to many other social systems and to analyse other kinds of discrimination.

References

1. Albertos, P., Benítez, I., Díez, J.L., Lacort, J.A.: Tool for equal opportunity evaluation in dynamical organizations. Proceedings of 13th IFAC Symposium on System Identification, SYSID-2003 (to be presented). Elsevier Science Ltd, Oxford, UK (2003)
2. Barberá, E., Albertos, P.: Fuzzy Logic Modeling of Social Behavior. *Cybernetics and Systems: an International Journal*, Vol 25, No.2 (1994) 343–358
3. Bezdek, J.C.: Pattern recognition with Fuzzy Objective Function Algorithms. Plenum Press (1987)
4. Centre for Gender and Diversity. University of Maastricht.
http://www.genderdiversiteit.unimaas.nl/frameset_uk.htm
5. Helbing, D.: Quantitative Sociodynamics. Kluwer Academic Publishers (1995)
6. Kimura, D.: Sex differences in the brain. *Scientific American*, Special Issue "The hidden mind", No. 12 (2002) 32–37
7. Modelling and Simulation of Ethnic and Social Process. <http://www.univer.omsk.su/MEP/>
8. Orborn, M., et al.: ETAN Report: Science Policies in the European Union: Promoting excellence through mainstreaming gender equality. Office for Official Publications of the European Communities, Luxembourg (2001)
9. Pantazopoulos, K.N., Tsoukalas, L.H., Bourbakis, N.G., Brün, M.J., Houstis, E.N.: Financial Prediction and Trading Strategies Using Neurofuzzy Approaches. *IEEE Transactions on Systems, Man and Cybernetics – Part B*, Vol. 28, No.4, (1998) 520–531
10. Pearson, D.W., Boudarel M-R.: Pair Interactions: Real and Perceived Attitudes. *Journal of Artificial Societies and Social Simulation*, Vol. 4, No. 4 (2001)
<http://www.soc.surrey.ac.uk/JASSS/4/4/4.html>
11. Shapiro, G., Olgiati, E.: Promoting gender equality in the workplace. Office for Official Publications of the European Communities (2002)
12. Stevens, I., Lamoén, V.I.: Garant Manual on Gender Mainstreaming at Universities. Chapter One (2001)
13. MIST – EWERC. The European Work and Employment Research Centre.
<http://www2.umist.ac.uk/management/ewerc/>
14. Wander, J., Popping, R., Van de Sande, H.: Clustering and fighting in two-party crowds: simulation of the approach – avoidance Conflict. *Journal of Artificial Societies and Social Simulation*, Vol. 4, No. 3 (2001) <http://www.soc.surrey.ac.uk/JASSS/4/3/7.html>

Approximate Solutions to Semi Markov Decision Processes through Markov Chain Montecarlo Methods

Arminda Moreno-Díaz¹, Miguel A. Virto¹, Jacinto Martín², and
David Ríos Insua³

¹ School of Computer Science
Madrid Technical University
{amoreno,mvirto}@fi.upm.es

² Dept. of Mathematics
University of Extremadura
jrmartin@unex.es

³ Statistics and Decision Sciences Group
Rey Juan Carlos University
drios@escet.urjc.es

Abstract. We explore the possibilities of Markov Chain Monte Carlo simulation methods to solve sequential decision processes evolving stochastically in time. The application areas of such processes are fairly wide, embedded typically in the Decision Analysis framework, such as preventive maintenance of systems, where we shall find our illustrative examples.

1 Introduction

Sequential decision making problems, in which a decision maker or agent chooses consecutive actions according to a system status and his preferences to form a decision policy, are present in a wide range of control and planning problems subject to stochastic effects and evolution. The essence of these problems is that decisions made now can have both immediate and long-term effects. The ultimate goal of these agent's actions would be maximizing the expected utility of such actions. Markov and Semi Markov Decision Processes (MDP and SMDP) provide a powerful mathematical representation of these problems.

We are interested in systems whose evolution can be modelled as Semimarkovian processes [9]. State transitions are stochastic and actions can be time dependent, state dependent or both. Our aim is to optimize these decisions or actions in a finite horizon, thus solving complicated dynamic decision problems with complex probabilistic environments evolving in time. Decisions are made sequentially, obtaining an immediate utility after each decision, which also modifies the environment for future decisions. This utility is a measure for preferences over consequences of actions. Other sources of complexity in the problems we consider, are that we include continuous decision or action spaces, continuous probability distributions or intricate relationships between its parameters.

Our aim is to explore the applicability of Markov Chain Monte Carlo (MCMC) simulation techniques to the solution of these optimization problems, see [6] for a review. The starting point will be the augmented probability method described in [1], which we improve to cope with sequences of interdependent decisions. Then, a backwards induction method will permit us to perform the maximization and marginalization steps required to solve the problem. The application area will be in systems' reliability and maintenance, see, for instance, [10], for a review.

The organization of the paper is as follows. Next section deals with the general structure of the problems we consider, embedded in the framework of semi Markov decision processes. Section 3 introduces MCMC simulation techniques which constitute the main part of the proposed algorithm, which is also presented in this section. Section 4 provides a theoretical proof of the proposed methodology. Section 5 illustrates the method with a maintenance example in which non-stationary policies are considered to maximize expected utility in a finite time horizon. We end up with conclusions and some discussion.

2 SMDP Framework

Our general modelling framework will be that of Semi Markov Decision Processes (SMDP). In such processes, decision periods are not restricted to discrete time ones as in MDPs. Decisions are made at specific system state change epochs relevant to the decision maker. These processes generalize MDPs in several ways, mainly by allowing actions to be history dependent and by modelling the transition time distribution of each action.

Formally, they are defined in terms of a set of states Θ , a set of actions or action space $\mathcal{A}$, a transition probability function P and a reward function. In our context, the latter will be represented by a utility function, $u(\cdot)$, that can depend directly on actions and states or on a random variable Y_t denoting consequences of those actions and states. For instance, y_t could designate the realization of a random variable 'income' in decision period $(t, t+1)$. The utility function stands for preferences over such consequences and it will be a generic function, possibly nonlinear but separable over decision epochs. Moreover, we impose that it must be a positive function.

Figure 1 shows the generic behavior of the system. The sequence of waiting times in the successive states are random variables with distribution depending on the states and on the previous states in the sequence. When a change of state occurs, a decision is made to control the process. A policy π is a decision-making function or control strategy of the agent, representing a mapping from situations to actions. Let θ_t be the state of the process at time t , including all random variables governing the system, random consequences and so on. A decision a_t is then chosen and, in random time τ_{t+1} , the process enters state θ_{t+1} .

Let a history of the process until time t be a random vector H_t such that for $t = 1$, $H_1 = (\Theta_0, \mathcal{T}_1, \Theta_1)$, and for $t \geq 2$, $H_t = (\Theta_0, \mathcal{T}_1, \Theta_1, \mathcal{A}_1, \mathcal{T}_2, \Theta_2, \mathcal{A}_2,$

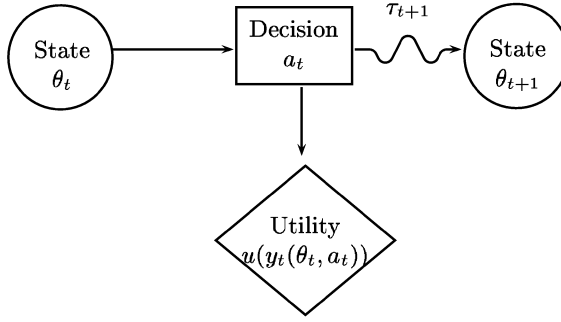


Fig. 1. Stages in a SMDP

$\dots, \mathcal{A}_{t-1}, \mathcal{T}_t, \Theta_t)$, where $\mathcal{T}_i$ denotes the i th transition time, Θ_i denotes the state space at this transition time and $\mathcal{A}_i$ the action space. The transition probability function of the system describes the probability that the system will be in state θ_{t+1} for the next decision period at or before τ_{t+1} time units since last decision was made, given the past history of the process.¹ We shall assume that this transition probability function verifies the markovian property, i.e.,

$$P(\tau_{t+1}, \theta_{t+1} | h_t, a_t) = P(\tau_{t+1}, \theta_{t+1} | \theta_t, a_t)$$

Further decomposition of this transition function is also possible. For instance, it could be computed as the product

$$P(\tau_{t+1}, \theta_{t+1} | h_t, a_t) = Q(\theta_{t+1} | \theta_t, a_t) F(\tau_{t+1} | \theta_t, a_t),$$

where Q describes the transition probability function at decision epochs and F gives the probability of the next transition time occurring τ_{t+1} units of time after action a_t has been taken in state θ_t . However, the complexity of the systems we are considering include no explicit or analytical representation for this transition kernel, allowing for a general class of models.

We may use mainly two criteria to control the system: infinite versus finite time horizon. In the former, we are interested in reaching a target state or in maximizing the expected utility regardless of the number of steps. In the latter, we are asking for the policy that maximizes expected utility in T time units or in a maximum number N of transitions. If this is the case, let $\nu_T^\pi(\theta)$ be the expected utility during T periods beginning from θ as the initial state and applying policy π . We are interested in finding the policy π that attains the maximum expected utility

$$\nu_T(\theta) = \max_{\pi \in \Pi} \nu_T^\pi(\theta), \quad \forall \theta \in \Theta$$

¹ It is important to note that we can also represent the process in terms of the time measured from the beginning until transition t has taken place.

where Π is the set of feasible policies. That goal can also be expressed in terms of a high-dimensional integral,

$$\pi^* = \arg \max_{\pi} \int \int u(\pi, \tau, \theta) p_{\pi}(\tau, \theta) d\theta d\tau \quad (1)$$

see [6] for further details. In (1) we can expand the vector (π, θ, τ) as the history $h_T = (\theta_0, \tau_1, \theta_1, a_1, \tau_2, \theta_2, a_2, \dots, \tau_t, \theta_t, a_t, \tau_{t+1}, \theta_{t+1})$ of relevant variables at each change point of the process, assuming the last decision made is a_T . We aim at solving problem (1), whose complexity may arise from complex probability structures, continuity of time, state and action space, intricate relationships between these elements and complex utility functions.

3 MCMC Simulation Methods

MCMC simulation methods constitute a powerful, versatile and accurate way to obtain samples from complex and non-standard distributions. As stated in [4] they can also be used to perform high dimensional integration tasks. However, as shown in [13] it is not always straightforward to perform those integrations when solving sequential optimization problems. For a general exposition of such methods see [7].

The algorithm we propose is of the Metropolis-Hastings kind and it makes use of the augmented probability method shown in [1]. With such method, we define an artificial distribution g on the product space $\mathcal{T} \times \Theta \times \mathcal{A}$, such that it is proportional to the function we want to integrate out and maximize in (1), i.e.,

$$g(\tau, \theta, a) \propto u(\tau, \theta, a) p_{\pi}(\tau, \theta).$$

In such a way the marginal distribution for a is given by,

$$\int \int g(\tau, \theta, a) d\theta d\tau \propto \int \int u(\tau, \theta, a) p_{\pi}(\tau, \theta, a) d\theta d\tau$$

and its modes are given by its maxima. Furthermore, the maxima of these two integrals will coincide. To finish the algorithm off we include a backward induction procedure performed on the sample resultant from the simulation output. Thus, we start generating a sample from the distribution $g(\tau, \theta, a)$ using MCMC methods. The algorithm goes as follows,

1. Start from an initial history

$$h^0 = (\tau_1^0, \theta_1^0, a_1^0, \dots, \tau_T^0, \theta_T^0, a_T^0)$$

Do $i = 0$.

1.1. Compute $u(h^0)$

2. Until convergence, iterate through

2.1. Generate a history h^c from distribution $q(\cdot|h^i)$

2.2. Compute $u(h^c)$

2.3. Compute

$$\alpha = \min \left\{ 1, \frac{u(h^c)p(\tau^c, \theta^c)q(h^i|h^c)}{u(h^i)p(\tau^i, \theta^i)q(h^c|h^i)} \right\}$$

2.4. Do

$$h^{i+1} = \begin{cases} h^c & \text{with probability } \alpha \\ h^i & \text{with probability } (1 - \alpha) \end{cases}$$

3. Use the sample $\{h^1, \dots, h^n\}$ to find the optimal action for each transition time and state with the backward induction procedure

Step 2 includes a distribution function q that supplies candidate values. Its choice will lead to specific and simpler versions of the algorithm that will simplify the computation of the α term, see e.g. [8] or [2]. In this computation it is necessary to impose $u(\cdot) > 0$. Step 3 finds the modes of the marginal distribution on decisions. Next section gives a detail explanation of this third step as well as a theoretical explanation of the complete method. We will illustrate its application in a maintenance decision making example.

4 Theoretical Proof

Suppose the artificial distribution $u(\tau, \theta, \pi)p_\pi(\tau, \theta)$ expanded in

$$u(\tau_1, \theta_1, a_1, \tau_2, \theta_2, a_2, \dots, \tau_T, \theta_T, a_T, \tau_{T+1}, \theta_{T+1})p_\pi(\tau_1, \theta_1, \dots, \theta_T, \tau_{T+1}, \theta_{T+1})$$

is known. From the conditional probability definition and the markovian property for states, assuming the independence of random variables τ and dependencies of θ only on the previous τ , we obtain the following decomposition

$$\begin{aligned} & u(\tau_1, \theta_1, a_1, \dots, \tau_T, \theta_T, a_T, \tau_{T+1}, \theta_{T+1})p(\tau_1)p(\theta_1|\tau_1)p_{a_1}(\tau_2|\theta_1)p_{a_1}(\theta_2|\tau_1, \tau_2, \theta_1) \dots \\ & \dots p_{a_{T-1}}(\tau_T|\theta_{T-1})p_{a_{T-1}}(\theta_T|\tau_T, \theta_{T-1})p_{a_T}(\tau_{T+1}|\theta_T)p_{a_T}(\theta_{T+1}|\tau_{T+1}, \theta_T). \end{aligned}$$

Let Θ_t , v_t and D_t be the vectors $(\theta_1, \theta_2, \dots, \theta_t)$, $(\tau_1, \tau_2, \dots, \tau_t)$ and $(a_1, a_2, \dots, a_t)$, respectively. Considering the integral in τ_{T+1} and θ_{T+1} , following last decision a_T , we define $U_T(a_T, D_{T-1}, v_T, \Theta_T)$ as

$$\int \int u(D_T, v_{T+1}, \Theta_{T+1})p_{D_T}(v_{T+1}, \Theta_{T+1})d\tau_{T+1}d\theta_{T+1}. \quad (2)$$

If we assume additive utilities, this utility can be decomposed in two terms,

$$u_T(D_T, v_{T+1}, \Theta_{T+1}) = u(D_{T-1}, v_T, \Theta_T) + u(a_T, \tau_{T+1}, \theta_{T+1})$$

and the integral in (2) would then be,

$$\begin{aligned} & p_{D_{T-1}}(v_T, \Theta_T) \left[u(D_{T-1}, v_T, \Theta_T) + \right. \\ & \left. + \int \int u(a_T, \tau_{T+1}, \theta_{T+1})p_{a_T}(\tau_{T+1}|\theta_T)p_{a_T}(\theta_{T+1}|\tau_{T+1}, \theta_T)d\tau_{T+1}d\theta_{T+1} \right] \end{aligned}$$

If history (D_{T-1}, v_T, Θ_T) has been fixed before last decision a_T , the first term is constant and the second one is proportional to the expected utility in the last stage, when decision a_T is taken. Thus, the maximum value for $U_T(a_T, D_{T-1}, v_T, \Theta_T)$ is the maximum expected utility of last decision a_T . Let a_T^* be the decision maximizing this expected utility

$$U_T^*(D_{T-1}, v_T, \Theta_T) = U_T(a_T^*, D_{T-1}, v_T, \Theta_T).$$

Being

$$E^*(\theta_T) = \int \int u(a_T^*, \tau_{T+1}, \theta_{T+1}) p_{a_T}(\tau_{T+1} | \theta_T) p_{a_T}(\theta_{T+1} | \tau_{T+1}, \theta_T) d\tau_{T+1} d\theta_{T+1}$$

we would have

$$\begin{aligned} U_{T-1}(a_{T-1}, D_{T-2}, v_{T-1}, \Theta_{T-1}) = \\ = \int \int (u(D_{T-1}, v_T, \Theta_T) + E^*(\theta_T)) p_{D_{T-1}}(v_T, \Theta_T) d\tau_T d\theta_T. \end{aligned}$$

Using again the additive decomposition for $u(D_{T-1}, v_T, \Theta_T)$, the last expression becomes

$$\begin{aligned} p_{D_{T-2}}(v_{T-1}, \Theta_{T-1}) \left[u(D_{T-2}, v_{T-1}, \Theta_{T-1}) + \right. \\ \left. + \int \int [u(a_{T-1}, \tau_T, \theta_T) + E^*(\theta_T)] p_{a_{T-1}}(\tau_T | \theta_{T-1}) p_{a_{T-1}}(\theta_T | \tau_T, \theta_{T-1}) d\tau_T d\theta_T \right]. \end{aligned}$$

Once more, if history $(D_{T-2}, v_{T-1}, \Theta_{T-1})$ has been formerly fixed, the maximum of

$$U_{T-1}(a_{T-1}, D_{T-2}, v_{T-1}, \Theta_{T-1})$$

is the maximum expected utility at time $T-1$, assuming that optimal action a_T^* has been taken in step T . Being a_{T-1}^* the decision that maximizes U_{T-1} and defining $E^*(\theta_{T-1})$ as

$$\int \int (u(a_{T-1}^*, \tau_T, \theta_T) + E^*(\theta_T)) p_{a_{T-1}}(\tau_T | \theta_{T-1}) p_{a_{T-1}}(\theta_T | \tau_T, \theta_{T-1}) d\tau_T d\theta_T$$

we continue to apply the same procedure on $T-2, T-3, \dots, 2$ and 1 where we would finally have

$$U_1(a_1, \tau_1, \theta_1) = p(\tau_1, \theta_1) \int \int (u(a_1, \tau_2, \theta_2) + E^*(\theta_2)) p_{a_1}(\tau_2 | \theta_1) p_{a_1}(\theta_2 | \tau_2, \theta_1) d\tau_2 d\theta_2.$$

Given (τ_1, θ_1) , the maximum for U_1 will provide optimal action a_1^* .

Being $u(\tau, \theta, \pi) p_\pi(\tau, \theta)$ the target distribution of the MCMC algorithm, the history frequencies obtained from the output, $fr(\tau_1, \theta_1, a_1, \tau_2, \theta_2, a_2, \dots, \tau_T, \theta_T, a_T, \tau_{T+1}, \theta_{T+1})$, give an approximation to that distribution. Accumulating the frequencies of all histories coinciding in all components except θ_{T+1} and τ_{T+1} , we shall have

$$fr(\tau_1, \theta_1, a_1, \dots, \tau_T, \theta_T, a_T) = \sum_{\tau_{T+1}} \sum_{\theta_{T+1}} fr(\tau_1, \theta_1, a_1, \dots, \tau_T, \theta_T, a_T, \tau_{T+1}, \theta_{T+1})$$

which, for each a_T , estimates integral (2), that is,

$$\hat{U}_T(a_T, D_{T-1}, v_T, \Theta_T).$$

The history with maximum frequency will contain an approximation to the optimal decision a_T^* , i.e.,

$$a_T^* = \arg \max_{a_T} \hat{U}_T(a_T, D_{T-1}, v_T, \Theta_T).$$

In the previous stage, we shall accumulate, for each a_{T-1}

$$fr(\tau_1, \theta_1, a_1, \dots, \tau_{T-1}, \theta_{T-1}, a_{T-1}) = \sum_{\tau_T} \sum_{\theta_T} fr(\tau_1, \theta_1, a_1, \dots, a_{T-1}, \tau_T, \theta_T, a_T^*).$$

Once more, these frequencies give an estimation,

$$\hat{U}_{T-1}(a_{T-1}, D_{T-2}, v_{T-1}, \Theta_{T-1})$$

and the history with maximum frequency includes optimal action a_{T-1}^* , the one satisfying

$$a_{T-1}^* = \arg \max_{a_{T-1}} \hat{U}_{T-1}(a_{T-1}, D_{T-2}, v_{T-1}, \Theta_{T-1}).$$

Repeating this accumulating procedure for previous stages, we will find optimal actions for partial histories, until last stage,

$$fr(\tau_1, \theta_1, a_1) = \sum_{\tau_2} \sum_{\theta_2} fr(\tau_1, \theta_1, a_1, \tau_2, \theta_2, a_2^*, \dots, a_T^*)$$

where the history with maximum frequency will give optimal action a_1^* , given previous history (τ_1, θ_1) .

5 Multi-component Maintenance Model

Designing a system maintenance policy and guaranteeing its operation is a delicate task. The improvements in analytical techniques and the availability of faster computers during the past two decades have allowed the analysis of more complex and realistic systems, hence the increasing interest in developing models for multi-component maintenance optimization. In [3] and [5], we can find a complete overview of such maintenance models and policies.

In addition to the conventional preventive and corrective maintenance policies, opportunistic maintenance arises as a category that combines the other two. Such policies refer basically to situations in which preventive maintenance (or replacement) is carried out at opportunities. It also happens that the action to be taken on a given unit or part at these opportunities depends on the state of the rest of the system.

In our example, these time opportunities will depend on the failure process and equal to breakdown epochs of individual components. The unpleasant event

Table 1. Example of MCMC algorithm output

τ_1	θ_1	a_1	τ_2	θ_2	a_2	τ_3	θ_3	a_3	τ_4	θ_4	a_4
1.808	0	1	3.070	3	2	9.000	0	9	9.000	0	9
2.934	4	1	2.956	3	1	3.324	2	1	3.840	1	1
2.055	4	1	3.450	3	1	3.489	2	2	9.000	0	9
3.130	4	1	3.768	3	1	3.858	2	1	3.993	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
3.435	4	1	3.758	3	1	3.781	2	1	3.895	1	2
3.562	4	1	3.598	3	1	3.658	2	1	3.799	1	2
3.130	4	1	3.768	3	1	3.858	2	1	3.993	1	1

of a failing component is at the same time considered as an opportunity for preventive maintenance on other non-failed, but deteriorated ones. The condition of a deteriorated or doubtful unit will be its age exceeding a critical given level. Thus, all components of the system are monitored and when one fails, there is complete information about the age of non-failed components and an opportunity to replace the whole system simultaneously. Hence, the two actions to be taken are: replace instantaneously only the failed component or replace the whole system.

5.1 Example

Assume we are managing a system consisting of five identical items (for instance machinery, electronic components, vehicles,...) for 4 years. Each item’s lifetime is modelled as a Weibull distribution with (7,4) as shape and scale parameters. An item older than two years is considered as a deteriorated one. Each time one of the items fails we count the number of deteriorated ones. This would be the state of the system, taking values 0, 1, 2, 3, 4. Then, we can choose whether to replace the only one that has failed or all the deteriorated. These actions will be denoted as 1 and 2, respectively. A replacement always leads to an ‘as good as new’ item and its duration is negligible. The decision epochs would then be the time a failure takes place. Due to the continuous time assumption, the Weibull distribution and this type of opportunistic maintenance model, there is no analytical form for the transition probability distribution P .

The cost and reward structure is simple. Breakdown of a component costs 20 monetary units (m.u.). A fixed cost of 10 m.u. is incurred for just one replacement operation whereas the cost for replacing the five units is 30 m.u. At the end of the fourth year, deteriorated items have lost 5 m.u. of its original value while non-deteriorated ones are worth the same. We also introduce a fixed reward of 120 m.u. in the course of these four years of operation to keep the final utility positive.

The output of the MCMC algorithm will comprise histories based on a continuous time scale, like the ones shown in Table 1. Sequence (9.000 0 9) indicates the end of the observed history.

Table 2. Example of MCMC algorithm output with time scale based on quarters

τ_1	θ_1	a_1	τ_2	θ_2	a_2	τ_3	θ_3	a_3	τ_4	θ_4	a_4	τ_5	θ_5	a_5	τ_6	θ_6	a_6	fr
7	0	1	14	3	1	99	0	9	99	0	9	99	0	9	99	0	9	51
8	4	1	9	3	1	11	2	2	99	0	9	99	0	9	99	0	9	3
8	4	1	14	3	1	14	2	1	15	1	1	99	0	9	99	0	9	15
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
9	4	1	11	3	1	14	2	1	15	1	1	15	0	1	99	0	9	4
9	4	1	12	3	1	99	0	9	99	0	9	99	0	9	99	0	9	104
9	4	1	13	3	1	13	2	1	99	0	9	99	0	9	99	0	9	22
9	4	1	13	3	1	15	2	1	15	1	1	99	0	9	99	0	9	35
10	4	1	15	3	1	99	0	9	99	0	9	99	0	9	99	0	9	938
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

To apply the backwards induction method on this history distribution, we need to discretise time. We will assume a time scale based on quarters, see Table 2. Once more, (99 0 9) indicates the end of the history. Thus, the time unit will range from $t = 0$ to $t = 15$. Given the parameters of the Weibull distribution, the expected time of failure is approximately 3.6, so the first histories recorded start with $t = 4$ and they show, at the most, six changes of state.

The structure of each history is the pattern (t, θ_t, a_t) repeated the number of times a failure has occurred. For instance, (7, 0, 1|12, 3, 1|13, 2, 2|13, 1, 1|15, 0, 2) means the following: at 7th quarter, an item failed, there were no others deteriorated and the action taken is to replace the only one broken. Then, at quarter 12th, another one failed, there were 3 more deteriorated and we decided to change only the broken one as $a_{12} = 1$. At quarter 13th, there were two items deteriorated and we decided to change the whole system. Still at this quarter, another one fails, but this time we decide to change the only broken, $a = 1$. At the last quarter, one item fails, there is no other deteriorated and we replace the broken one.

As we note in this example, for a given time it is also possible that more than one event occurs. This is produced by the lifetime distribution for components, the time scale chosen or the sample size and it should be taken into account when designing a proper backward induction algorithm. We will illustrate a suitable one for this example. We start from $t = 15$, looking for all histories with $(15, \cdot, \cdot)$ as the final component. Aggregating its frequencies we obtain Table 3.

If we were to solve decision in time $t = 15$ based only on this table this is what happens. For state $\{0, 1, 2, 3\}$, we will choose $a = 1$ and for state 4, $a = 2$. But if we look for those histories containing $(15, 4, 1)$ in the last but one position followed by $(15, \cdot, a^*)$, we find such histories having a frequency of 5367. Aggregating that figure to 19318, we conclude that the optimal action for $t = 15$ and $\theta_t = 4$ is also $a_t = 1$.

This means that before we solve a decision for a given time and state, we must check that all the frequencies of all histories having that pattern in another position, and followed by optimal patterns recently solved, confirm the action

Table 3. First aggregation attempt

t	x_t	a_t	
		1	2
15	0	87	39
15	1	2473	1329
15	2	12035	9564
15	3	25369	24479
15	4	19318	22819

Table 4. Final Table

State Decision		Time quarter									
		16	15	14	13	12	11	10	9	8	7
0	1	87	18	0	0	0	0	0	0	439	161
	2	39	4	0	0	0	0	0	0	492	161
1	1	2537	542	68	9	0	0	0	0	0	0
	2	1329	288	31	12	0	0	0	0	0	0
2	1	13359	4612	1166	227	45	2	0	0	0	0
	2	9564	3522	851	144	33	6	3	0	0	0
3	1	30693	18741	9073	3221	1034	256	52	7	0	0
	2	24479	16131	7898	3117	857	285	78	10	0	0
4	1	26136	28186	24022	17605	10850	5949	2787	1217	0	0
	2	22819	27133	24336	17922	11244	6236	3058	1331	0	0

taken. Then, each time we solve the decision for a given state we should update the frequencies’ distribution, cancelling those histories not having the optimal action for the given state and shortening those with optimal actions, cancelling the pattern (t, θ_t, a_t^*)

We start with histories having $(15, 0, \cdot)$ in the last position. As the frequency for $(15, 0, 1)$ is 87 and for $(15, 0, 2)$ is 39 we would choose $a = 1$ as optimal decision. Before concluding that, we must check for those histories with that pattern in the last but one position. As there are not any, when we are in time $t = 15$ and state $\theta_t = 0$ the optimal action is to change only the failed item. Then, we cancel all histories ending with $(15, 0, 2)$ as they are not optimal and shorten those ending with $(15, 0, 1)$, erasing that part. We now look for histories with pattern $(15, 1, \cdot)$ in the last position and find $(15, 1, 1)$ with frequency 2537 and $(15, 1, 2)$ with frequency 1329. The former does not coincide with the one in Table 3 because those ending with $(15, 1, 1|15, 0, 1)$ are also included as a result of the update made when solving previous decision. If we find the sequence $(15, 1, \cdot)$ followed by other pattern that has not been solved yet, we would try to solve the latter first. If this is not possible, we cannot go on and the algorithm must stop. In that case, we need to modify the time scale.

As this does not happen here, we choose $a = 1$ when $t = 15$ and $\theta_t = 1$ and update the frequency distribution. We follow with $t = 15$ and $\theta_t = 2$ and so on filling in Table 4. This table shows in boldface maximum frequencies and optimal actions in the first ten quarters of our study, for each possible state.

For instance, being in quarter 14th and observing 3 deteriorated machines after one has failed, we should decide to replace the only one broken as the optimal action.

Solution provided by the algorithm is consistent in the following sense. If for a given decision epoch or quarter t and state θ_t the optimal action is to replace the whole system, this decision remains optimal for greater values of θ_t in the same quarter. This is the case at $t = 10$ and $t = 11$ where optimal action is $a = 2$ when $\theta \geq 3$. Additionally, as time goes by, optimal action is to replace only the broken component.

6 Discussion

This paper presents a simulation method to cope with the complexity inherent to sequential decision making problems when intricate stochastic scenarios are present. In these scenarios, the objective is to maximize expected utility of an agent's decisions in a non-stationary environment evolving in time. The mathematical complexity of its formulation leads to simulation methods as a suitable way to approximate optimal solutions. Therefore, we have explored the possibilities of MCMC simulation methods combined with the augmented probability method and a backward induction procedure to obtain approximate solutions. This methodology has been successfully applied to solve markovian optimization problems [13] and influence diagrams [12]. This paper illustrates its generalization in wider environments.

The methodology presented is not designed to reduce dimensionality of the problem since the proposed algorithm handles the whole histories, affecting the computational burden. However, there are other potential difficulties in stochastic dynamic programming such as the explicit knowledge of transition probability functions, required by classical resolution methods. This issue could be ever more critical and demanding than the dimensionality itself.

Interesting future research directions include non-markovian processes, partially observed environments or interaction of multiple agents. The first one is outlined in [11] together with other examples of the methodology proposed herein.

Acknowledgements. Research supported by projects from CAM, URJC and MCYT (DPI-2001-3731).

References

1. Bielza, C., Müller, P., Ríos-Insua, D.: Decision Analysis by augmented probability simulation. *Management Science*, 45 (1999) 1552–1569.
2. Chib, S., Greenberg, E.: Understanding the Metropolis Hastings Algorithm. *The American Statistician*, 49, 4 (1995) 327–335.
3. Cho, D.I. and Parlar, M.: A Survey of Maintenance Models for Multi-units Systems. *European Journal of Operational Research*, 51 (1991) 1–23.

4. de Freitas, N., Hojen-Sorensen, P., Jordan, M.I. and Russell, S.: Variational MCMC. In J. Breese and D. Koller (Ed.). *Uncertainty in Artificial Intelligence, Proceedings of the Seventeenth Conference* (2001)
5. Dekker, R., Van der Duyn Schouten, F. and Wildeman, R.: A Review of Multi-Component Maintenance Models with Economic Dependence. *Mathematical Methods of Operations Research*, 45 (1997) 411–435.
6. French, S., Ríos Insua D.: *Statistical Decision Theory*, Arnold (2000)
7. Gamerman, D.: *Markov Chain Monte Carlo: stochastic simulation for Bayesian inference*. Chapman & Hall (1997)
8. Hastings, W.K.: Monte Carlo Sampling Methods Using Markov Chains and its Applications. *Biometrika*, 57 (1970) 97–109.
9. Puterman, M. L.: *Markov Decision Processes*, In *Handbooks in OR & MS*, Vol. 2. Elsevier (1990)
10. Shaked, M., Shantikumar, J. G.: *Reliability and Maintainability*, In *Handbooks in OR & MS Vol.2*. Elsevier (1990)
11. Virto, M.A.: *Métodos Montecarlo en Análisis de Decisiones*. PhD. Thesis. Universidad Nacional de Educación a Distancia. Madrid (2002)
12. Virto, M.A., Martín, J., Ríos Insua, D. and Moreno-Díaz, A.: Approximate Solutions of Complex Influence Diagrams through MCMC Methods. In *Proceedings of First European Workshop on Probabilistic Graphical Models*. Gámez and Salmerón (Eds.) (2002) 169–175.
13. Virto, M.A., Martín, J., Ríos Insua, D. and Moreno-Díaz, A.: A Method for Sequential Optimization in Bayesian Analysis. *Bayesian Statistics*, 7 (J.M. Bernardo, M.J. Bayarri, J.O. Berger, A.P. Dawid, D. Heckerman A.F.M Smith and M. West eds.) Oxford University Press (2003) 701–710.

Knowledge Base for Evidence Based Medicine with Bioinformatics Components

Witold Jacak¹, Karin Pröll¹, and Jerzy Rozenblit²

¹ Department of Software Engineering,
Upper Austria University of Applied Sciences,
Hagenberg, Austria

² Department of Electrical and Computer Engineering
University of Arizona,
Tucson, USA

Abstract. This paper presents an approach for a multilevel knowledge base system for evidence-based medicine. A sequence of events called patient trial is extracted from computer patient records. These events describe one flow of therapy for a concrete disease. Each event is represented by state and time. We introduce a measure between states, which is used to calculate the best alignment between different patient trials. The alignment measure calculates the distance between two sequences of patient states, which represents the similarity of the course of disease. Based on that similarity- value classes are introduced by using specific clustering methods. These classes can be extended by gene expression data on micro-arrays leading to finer clustering containing similar trials - called trial families. For easy checking if a new trial belongs to a family we use profiles of Hidden Markov models to detect potential membership in a family.

1 Introduction

Knowledge based system techniques and applications will be one of the key technologies of new medicine approaches. A variety of intelligent techniques have been initiated to perform intelligent activity such as diagnosis. Among them knowledge based techniques are the most important and successful branch. Especially, in new clinical information systems (CIS), called evidence based medical systems (EBM) the construction of an appropriate knowledge base is one of the most important problems. Evidence based medicine/healthcare is looked upon as a new paradigm, replacing the traditional medical paradigm, which is based on authority. It depends on the use of randomised controlled trials, as well as on systematic reviews (of a series of trials) and meta-analysis, although it is not restricted to these. There is also an emphasis on the dissemination of information, as well as its collection, so that evidence can reach clinical practice. It therefore has commonality with the idea of research-based practice [3], [4], [5].

Evidence based medicine is the integration of *best research evidence* with *clinical expertise* and *patient values*. (see Fig.1.)

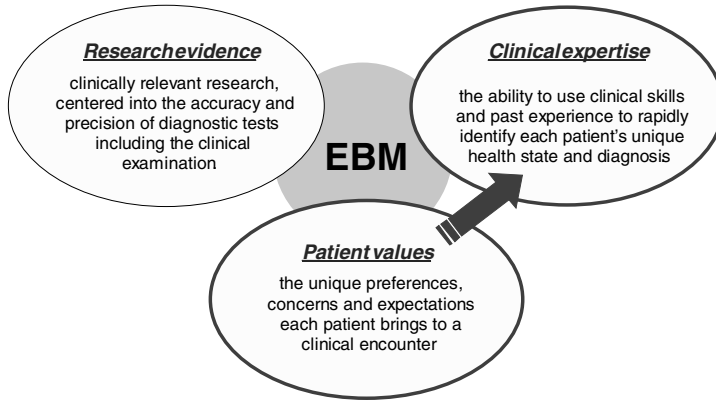


Fig. 1. Evidence based medicine (EBM) is the integration of best research evidence with clinical expertise and patient values.

- by *best research evidence* we mean clinically relevant research, often from the basic sciences of medicine, but especially from patient centered clinical research into the accuracy and precision of diagnostic tests (including the clinical examination), the power of prognostic markers, and the efficacy and safety of therapeutic, rehabilitative, and preventive regimens. New evidence from clinical research both invalidates previously accepted diagnostic tests and treatments and replaces them with new ones that are more powerful, more accurate, more efficacious, and safer.
- by *clinical expertise* we mean the ability to use our clinical skills and past experience to rapidly identify each patient's unique health state and diagnosis, their individual risks and benefits of potential interventions, and their personal values and expectations.
- by *patient values* we mean the unique preferences, concerns and expectations each patient brings to a clinical encounter and which must be integrated into clinical decisions if they are to serve the patient.

When these three elements are integrated, clinicians and patients form a diagnostic and therapeutic alliance, which optimises clinical outcomes and quality of life.

This article focuses the consideration on the multilevel approach for constructing an evidence based knowledge system for classification of different therapies and their effectiveness for clinical trials.

2 Multilevel EBM Knowledge Base System

In the first phase, a sequence of events called *patient trial* will be extracted from computer patient records (CPR). These events describe only one flow of therapy of a concrete disease. Each event is represented as a pair (*state*, *time*). The *state* contains not only standard numeric parameters but also images (MR, RT, or photo) and text based linguistic descriptions. Based on such *state* we introduce the measure between

states of different patient. We assume that each patient's state is represented in the global state space. Based on the measure the system calculates the best alignment between different patient trials. The alignment measure (score) calculates the distance between the two sequences of patient states, which represents the similarity of the flow of the therapy [2].

This procedure is applied to each pair of patient trials stored in the Hospital Information Systems (HIS) concerning similar diseases. Based on the value of similarity a semantic network is constructed and divided into full-connected partitions. Each of these partitions represents the class of the similar therapy and can be used for computer-aided decision-making in evidence based medicine diagnostic. The clustering only performed on the basis of medical data can be extended by the additional clustering process based on gene expression data for these same patients sets. The general structure of the system is sketched in the Fig.2.

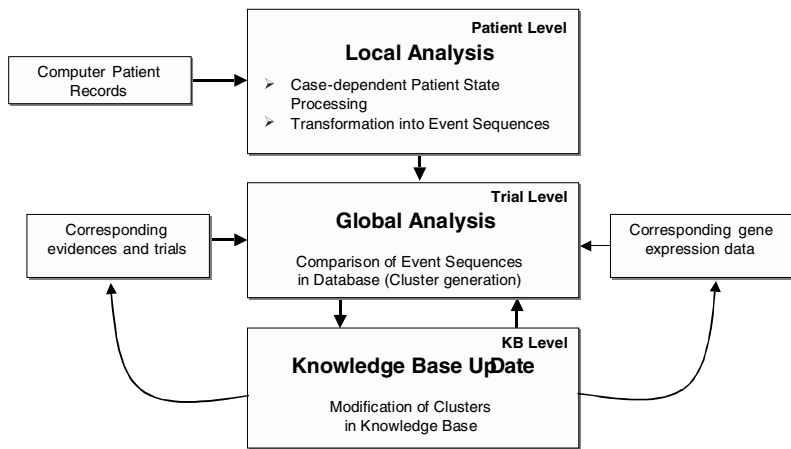


Fig. 2. Patient Values Knowledge Base for Evidence Based Medicine

3 Patient Record Level

On the patient level of knowledge base the data form patient records should be preprocessed to obtain the compact representation of course of disease. Normally we have different sources of medical information concerning a couple of numerical data representing the labor test results, some RT images, linguistic text describing diagnosis and therapy. The general patient record can be described as

$$Patient\ Record = \{(v_j | j=1, \dots, n), \{image_1, \dots, image_n\}, diagnosis, therapy\}$$

In the first phase the patient record should be transformed into sequence of events e representing the time course of a patient healing process called patient trials.

$$Trial = (e_1, \dots, e_n) = ((state_1, time_1), \dots, (state_n, time_n))$$

In order to compare the different trials it is necessary to introduce a formal description of states to allow the calculation of the distance or similarity between two states. It is

obvious that the vector of numerical data ($v_j | j=1, \dots, n$) is easy to compare. To make it possible to measure the similarity between images we propose to map images with respective 3D models which can be used to obtain additional numerical data ($w_j | j=1, \dots, k$) describing images. Such methods are very useful for dermatological diseases.

Based on 3D models we cannot only calculate various geometrical parameters as for example field, contour length or shape but we also can automatically perform a classification of infected anatomical regions and a coding of disease (Fig. 3.). Additionally we assume that diagnosis will be transformed into standard code for example ICD-10.

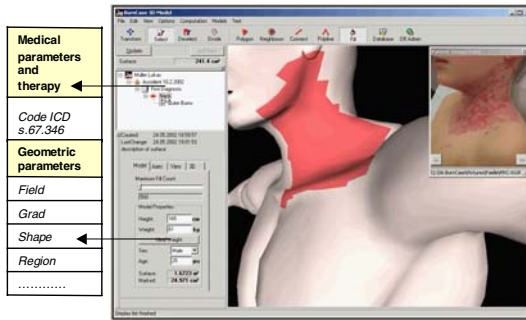


Fig. 3. Case-dependent Patient State containing numerical data and 3D-model data

After these preprocessing steps we can represent the state as a vector of numerical parameters and codes.

$$e^i = (state^i, time^i) = (((v_j^i | j=1, \dots, n), (w_j^i | j=1, \dots, k), diag\_code^i), time^i)$$

Based on such *state* we introduce the measure of distance

$$\rho : S \times S \rightarrow \mathbb{R}^+$$

between states of different patient (see Fig.4.).

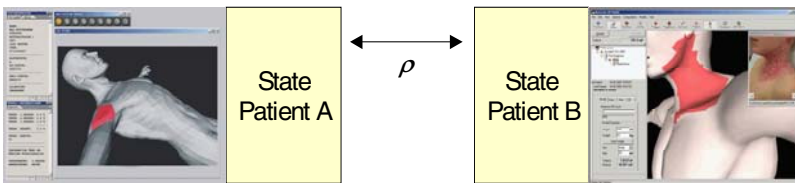


Fig. 4. The local measure between states of different patient.

We assume that each patient's state is represented in this global state space S . This distance represents the similarity of two states from different trials of the course of disease. An example of a patient trial for a dermatological disease is presented in Fig.5.

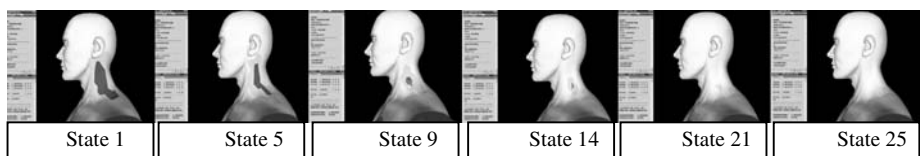
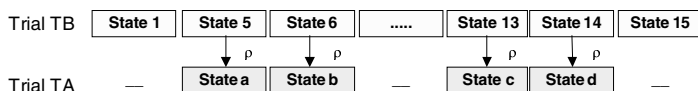


Fig. 5. Example of a patient trial

The images above are extracted from a concrete patient trial showing the healing process of a dermatological case in the neck region. Each state is based on a 3D model of the affected part of the body and a set of numeric and code parameters representing results of assessments or treatment protocols. The 3D Model and the parameters are used to build the state vector at each point of time, which are used to calculate the measure between states of treatment in different patient trials.

4 Trial Level : Alignment of Trials

The most basic trials analysis task is to ask if two trials are related. There are many positions at which the two corresponding patient states (residues) are very similar. However, in the case when the one of the trial has extra residues, then in a couple of places gaps have to be inserted to the second trial to maintain the alignment across such regions. When we compare sequences of the patient states, the basic processes are considered as substitution, which changes residues in the sequence, and insertion or deletion, which add or remove residues. Insertions and deletions are referred to as gaps. An example for state alignment can be seen in Fig.6.



$$\rho^* = \text{minimal total score} = \text{optimal alignment}$$

Fig. 6. Alignment between two trials

Based on the distance ρ the system calculates the alignment score ρ^* which describes the similarity (distance) between different patient trials.

$$\rho^* : S^* \times S^* \rightarrow R^+$$

where S^* is the set of ordered sequences of the states from the state space S . Each sequence

$$x = (x_1, \dots, x_n) \in S^*$$

has the respective events sequence describing the concrete patient trial $(e_1, \dots, e_n)$. The measure $\rho^*(x, y)$ calculates the distance between these two sequences, which describes the similarity of the flow of the therapy. For solving this problem we use a special sequence-matching algorithm.

The global measurement (score) we assign to an alignment will be the sum of terms for each aligned (similar) pair of states (residues), plus terms of each gaps (see Fig.7.). We will consider a pair of trials (patient states sequences) x and y of lengths n and m , respectively.

Let x_i be the i th state in x and y_j be the j th symbol of y . Given a pair of trials, we want to assign a score to the alignment that gives a measure of the relative likelihood that the trials are related as opposed to being unrelated. For each two states x_i and y_j we can use ρ as measure of similarity of this residues. Let $n > m$ then we should add the gaps g in the second trial to find the best alignment.

4.1 Gap Penalties

We expect to penalize gaps. Each state x_i in the sequence x has additional parameter τ_i which represent time interval between the state x_{i-1} and x_i . The first state x_1 has $\tau_1=0$. For finding the penalty value of gap at the i th position of the trial y we use the knowledge about time intervals associated with each state. Let the last ungaped substitution with state y_l has place on $(i-k)$ th position in the trial x .

x_{i-k}	x	x	x_{i-l}	x_i
y_l	g	g	g	g

Fig. 7. Gaps Insertion

The standard cost associated with a gap is given by

$$\rho(x_i, g) = K \exp(-(|\tau_i^x - \tau_l^y|)) = \text{pen}(x_i, y_l)$$

where τ_i^x is the time interval associated with state x_i and τ_l^y is the time interval associated with state y_l from trial y which was aligned with the state x_{i-k} from the trial x .

The long insertions and deletions with different intervals of time are penalized less as those where the intervals of the time is quite the same.

4.2 Alignment Algorithm

The problem we consider is that of obtaining optimal alignment between two patients trials, allowing gaps. The problem can be defined as follows:

Find the best alignment between sequence x^* and y^* (x^* , y^* represent sequences x and y extended of necessary gaps) such that global score

$$\rho^*(x, y) = \Sigma(\rho(x_i^*, y_i^*)) = \min$$

where $x_i^* = x_i$ or gap g and

if x_i is aligned to y_l and x_{i+k} is aligned to y_u then $l < u$.

We can use the known dynamic programming algorithm, which has many applications in the biological sequences analysis [1].

The idea is to build up an optimal alignment using previous solutions for optimal alignments of smaller subsequences.

We construct a matrix F indexed by i and j , one index for each trial, where value $F(i,j)$ is the score of the best alignment between the initial segment $x_1, \dots, x_i$ of x up to x_i and the initial segment $y_1, \dots, y_j$ of y up to y_j . We can build $F(i,j)$ recursively. We begin by initializing $F(0,0) = N$ (N is the large number). We then proceed to fill the matrix from top left to bottom right. If $F(i-1,j-1)$, $F(i-1,j)$ and $F(i,j-1)$ are known it is possible to calculate $F(i,j)$.

Let us assume that we are only interested in matches scoring ρ less than some threshold T , it means that similarity between two states of the patients is very high. $F(i,0)$ is the minimal (best) sum of completed match scores to the subsequence $x_1, \dots, x_i$ assuming that x_i is in an unmatched region.

It is clear that we expect that one trial contains the other or that they overlap. It means that we want a match to start on the top or left border of the matrix and finish on the right or bottom border. The initialization equations are that $F(i,0) = 0$ for $i=1, \dots, n$ and $F(0,j) = 0$ for $j=1, \dots, m$. Now we calculate recursively the matrix value as

$$F(i,0) = \min \begin{cases} F(i-1,0) \\ F(i-1,m)+T \end{cases}$$

$$F(i,j) = \min \begin{cases} F(i-1,j-1) + \rho(x_i, y_j) \\ F(i-1,j) + \text{pen}(x_{i-1}, y_j) \\ F(i,j-1) + \text{pen}(x_i, y_{j-1}) \end{cases}$$

The calculation steps are presented in Fig.8.

	TB1	TB2	TB3	
TA1	$F(i-1, j-1)$	$F(i, j-1)$	14	19
TA2	$F(i-1, j)$	$F(i, j)$	21	16
.....	15	13	12	14

1) Score calculation 2) Traceback

Fig. 8. Matrix for alignment calculation

Let $F_{\min}$ be the minimal value on the right border (i,m) for $i=1, \dots, n$, and the bottom border (n,j) $j=1, \dots, m$. This minimal score is the measure of the similarity between the complete two trials x and y . i.e.

$$\rho^*(x,y) = F_{\min}$$

To find the alignment itself we must find the path of choices that led to the minimal value. The procedure for doing this is known as a traceback. The traceback starts from the minimal point and continues until the top or left edge is reached.

5 Clustering of the Trial Space

Based on the alignment score ρ we can define similarity classes on the set of trials. The similarity class C is defined as follows:

- $C \subset \text{Set of Trials}$
- $(\forall x, y \in C)(\rho(x, y) < \varepsilon)$
- $\text{card}(C) \rightarrow \max$

where ε is the threshold value discriminating the similarity of two trials. For construction of similarity classes we build the graph which nodes are trials and its arcs is the relation ρ . The threshold value cuts different arcs. The similarity class is the maximal subgraph of the graph for which its nodes are full connected (see Fig.9.). There are many algorithms for constructions similarity classes [7].

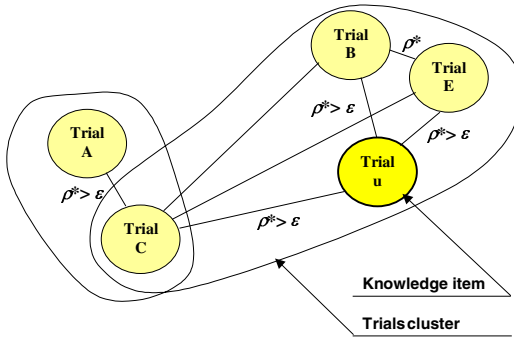


Fig. 9. Trials clustering

6 Gene Expression Clustering

As an additional support for diagnostics gene expression data can be used. Using micro-arrays, we can measure the expression levels of more genes simultaneously. These expression levels can be determined for samples taken at different time points during a biological process or for samples taken under different conditions. For each gene the arrangement of these measurements into a vector leads to what is generally called an expression profile. The expression vectors can be regarded as data points in high dimensional space. Cluster analysis in a collection of gene expression vectors aims at identifying subgroups (clusters) of such co-expressed genes, which have a higher probability of participating in the same pathway. The clusters can be used to validate or combine the cluster to prior medical knowledge. Many clustering methods are available, which can be divided into two groups: first and second generations algorithms. The first generation algorithms are represented by hierarchical clustering algorithms, K-means clustering algorithms or self-organizing maps. These algorithms are complicate in use and often require the predefinition of more parameters that are hard to estimate in biomedical praxis. Another problem is that first generation

clustering algorithms often force every data to point into a cluster. It can lead to lack of co-expression with other genes. Recently new clustering algorithms have started to tackle some of limitations of earlier methods. To this generation of algorithms belong: Self-organizing tree algorithms, quality based clustering and model-based clustering [2], [6], [8], [9]

Self-organizing tree algorithms combine both: self-organizing maps and hierarchical clustering. The gene expression vectors are sequentially presented to terminal nodes of a dynamic binary tree. The greatest advantage is that the number of clusters does not have to be known in advance. In quality based clustering clusters are produced that have a quality guarantee, which ensures that all members of cluster should be co-expressed with all members of these cluster. The quality guarantee itself is defined as a fixed threshold for a maximal distance between two points between clusters. Based on these methods it is possible to generate the clusters on the gene expression states.

Let G be a gene expression cluster, which contains the data obtained from micro-arrays. Each micro-array is connected with one or more patient trials. The cluster G can be transformed as a cluster GT to the patient trial space. This results in two patterns, which can be used for classifying each trial. On the one side the pattern based on trial alignment - on the other side the pattern based on gene expression can be used. Both patterns can be combined for creation fine classes containing very similar trials with high co-expression of genes [9].

7 Profiles of Patient Trials Families

In the previous sections we have already created a set of trials belonging to a particular cluster. Such a cluster is called a trial family. Trials in a family are diverged from each other in the primary sequence of the course of a disease. For easy checking if a new trial belongs to a family we propose to use statistical features of the whole set of trials in the cluster. With such a description it is possible to decide how strong the given trial is related to a cluster. We will develop a particular type of a Markov model, which is well suited for modeling multiples alignment in a group [1].

The profile of the trials family with maximal length n should describe the probability of an observation of a specific state of a patient in i -th position of the trial. It means that the profile can be formalized as:

The profile P of length n based on states set S is the matrix

$$P = [\pi_i(s)] \ i = 1, \dots, n \text{ and } s \in S$$

of probabilities. $\pi_i(s)$ is the probability that s occurs on position i in the sequence of states.

The approach is to build a hidden Markov model with a repetitive structure of states but different probabilities in each position. The key idea behind profile HMM is that we can use the same structure, but set the transition and emission probabilities to capture specific information about each position in the whole family of trials. The model represents the consensus sequence for the family, not the sequence of any particular member.

A probabilistic model would be to specify probabilities $\pi_i(s)$ of observing the state s in position i of trial. Such a model is created by Hidden Markov models with insertion and deletion where for each matched state the emission distribution for each patient state is estimated - for insert regions the emission and transition distribution is estimated and for deletion only the transition one. One of the main purposes of developing profiles of Hidden Markov models is to use them to detect potential membership in a family by obtaining significant matches of a given trial to a profile.

To find out if the observation trial belongs to the trials family we look for the most probable path in the Hidden Markov Model in the family. The most probable path in the HMM can be found recursively with standard Viterbi algorithm. Using this algorithm we can calculate the maximal probability a the given trial belongs to the cluster. The most difficult problem faced when using HMM is specifying the model in the first place. There are two parts to do this: the design of the structure, i.e. what states there are and how they are connected and the assignment of the parameter values, the transition and emission probabilities. The estimation of these parameters can be performed by using the Baum-Welch training algorithm. The choice of the length of the model corresponds more precisely to the decision on which multiple alignment columns in trial family to assign to match states, and which to assign insert states. A simple rule working well is that columns that are more than half gaps should be modeled by inserts.

The family profile can be used to predict the course of the disease based only on an initial subsequence of the trails. In this sense the profiles are very important tools in decision support for medical therapy.

8 Final Remarks

Evidence-based Systems gain a more important role in decision support for medical praxis. This paper presents a new approach for a multilevel knowledge base system for evidence-based medicine combined with bioinformatics components. In the first level a sequence of events called patient trial is extracted from computer patient records. These events describe one flow of therapy for a concrete disease. Each event is represented by state and time. We introduce a measure between states, which is used to calculate the best alignment between different patient trials. The alignment measure calculates the distance between two sequences of patient states, which represents the similarity of the course of disease. On the second level based on that similarity-value classes are introduced by using specific clustering methods. These classes can be treated more exactly by combining the information about gene expression data on micro-arrays. This leads to finer clustering containing similar trials - called trial families. For easy checking if a new trial belongs to a family we use profiles of Hidden Markov models to detect potential membership in a family. This knowledge base can be used to support diagnostic in hospital praxis.

References

1. Durbin R., Eddy S., Krogh A. Mitchison G.: Biological sequence analysis, Cambridge University Press (1998), UK
2. Moreau Y., De Smet F., Thijs G., Marchal K., De Moor B.: Functional Bioinformatics of Microarray Data: From Expression to Regulation, Special Issue on: Bioinformatics, Part I, Proceedings of the IEEE, Vol. 90, Number 11, (2002), 1722–1743
3. Gosling A. S., Westbrook J. I., Coiera E. W.: Variation in the use of online clinical evidence: a qualitative analysis, International Journal of Medical Informatics, 69; 1, (2003), 1–16.
4. Warner HR, Sorenson DK, Bouhaddou O. : Knowledge engineering in health informatics. Springer-Verlag, New York, NY. (1997)
5. Friedman CP, Wyatt JC.; Evaluation Methods in Medical Informatics. Springer, New York (1997).
6. Kohonen, T. Self-organizing Maps, Springer-Verlag, (1997), Berlin, Germany.
7. Kaufmann L., Rousseeuw P.J. Finding groups in Data: An Introduction to Cluster Analysis, Wiley, (1990), New York, USA
8. Moreau Y., De Smet F., Thijs G., Marchal K., De Moor B. : Adaptive quality-based clustering of gene expression profiles, Bioinformatics, Vol.18, no.5 (2002)
9. Aris V, Recce M.: A method to improve detection of disease using selective expressed genes in microarray data, Methods of Microarray Data Analysis , eds. SM Lin, KF Johnson (Kluwer Academic) (2002), pp. 69–80

Diversified Approach to Methodology and Technology in Distributed Intelligent Building Systems

Andrzej Jablonski¹, Ryszard Klempous¹, and Benedykt Licznarski²

¹ Faculty of Electronics

² Faculty of Microsystem Electronics and Photonics
Wrocław University of Technology,
27 Wybrzeża Wyspiańskiego str,
50-370 Wrocław, Poland

Abstract. The conception of Intelligent Building was created in North America by building-automation systems designers and manufacturers. It was used to describe buildings where microprocessor-based technologies were used in order to improve the buildings' efficiency. This paper discusses the basic issues related to the design of an intelligent building. It offers various solutions to the issues of data acquisition and processing, including the most important ones that are based on the distributed building automation system.

1 Introduction

Process automation has for a few years been providing solutions to a new field of applications - the building industry. Such important issues include: the minimization of energy consumption, the ensuring of comfort and safety to users, the lowering of environmental impact and the flexible allocation of inner space.

To fulfill these requirements electronic systems to control building functions needed to be introduced. It was necessary to integrate all the systems into one extensive system in order to co-ordinate their operation. Data acquisition, control and processing processes occurring in such a system require appropriate technical infrastructure. In a wider sense, owing to an integrated computer system, the concept of the intelligent building suggests a capability to exercise full control not only of the building and its services but also of the operation of organizations that are housed in the building.

2 Structures of Data Acquisition and Control Systems

Two categories of systems occur in the intelligent building. The first covers functional (generic) systems that meet the basic requirements of ensuring building safety, managing energy distribution, controlling comfort, arranging

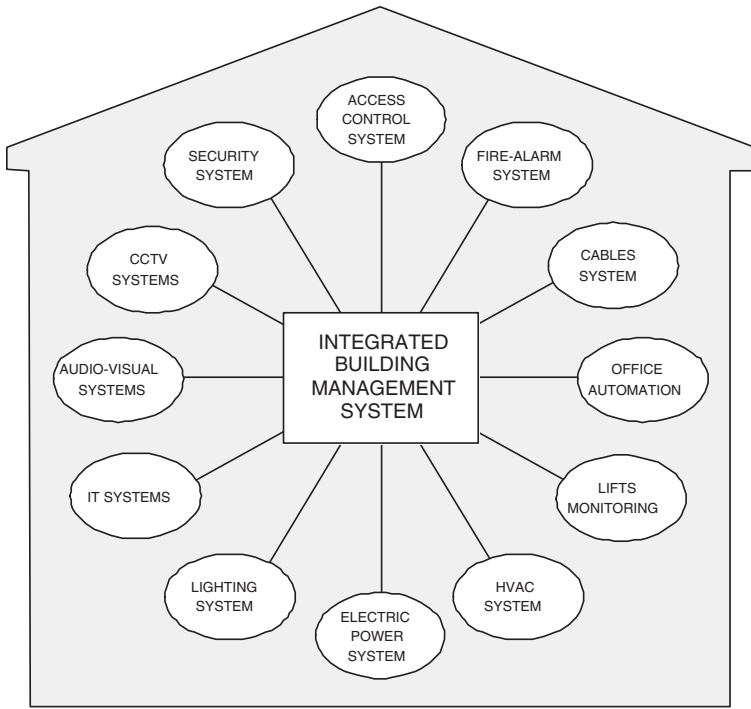


Fig. 1. Shows a typical functional structure of the intelligent building. The most frequent functional subsystems are highlighted. Such subsystems can be extended or constrained in every building

information and data flow, etc. The second one covers systems for taking measurements and acquiring data, processing information for control and management applications and reporting, archiving and visualizing processes within the building.

Looking back, early classic automation was based on independent control circuits (loops) and afterwards on centralized systems with a single high-capacity computer. The history of building automation remembers non-processor based independent functional systems (for example, fire detection, closed-circuit television or access control systems). Later the systems were equipped with built-in processors to process information and exercise control locally. These, however, were not integrated into a single system.

Uniform integrating solutions occurred in process and building automation along with the introduction of distributed information processing, where local systems (or intelligent subsystems) were linked via communication interfaces into a single global system based on supervisory processors. The assessment of real capacities for creating such structures is one of the objectives of this paper.

3 Characteristics of Computations in Building Automation Systems

Different algorithms are applied to every generic (functional) subsystem. Each subsystem receives different input data and generates output signals for different purposes. For example, a fire detection system receives data from smoke and flame detectors. It sends return signals to control extinguishing equipment, sound signalling devices, lift operation, etc. The audio and video presentation system receives audio and video signals, processes them and uses the data to control lighting, projectors, loudspeakers, etc. This means a decided dissimilarity between information processing algorithms and their technical solutions. This also means a natural need for processing information locally by local processors. The peculiarities of processing in intelligent building subsystems will be taken up in this paper.

In order to benefit globally from the system integration, or the synthesis of the intelligent building, a few most important information-processing algorithms need to be adapted. They are as follows:

The SECURITY and SAFETY (S&S) algorithm

$$S\&S = f \left(AccessControlSystem, FireAlarmSystem, \right. \\ \left. SecuritySystem, CCTV, LightingSystem \right)$$

The COMFORT and LIGHT (C&L) algorithm

$$C\&C = f \left(HVACSystem, HeatingSystem, SanitarySystem, \right. \\ \left. LightingSystem, AccessControlSystem, time \right)$$

The REDUCTION OF ENERGY CONSUMPTION (REnC) algorithm

$$REnC = f \left(Comfort, LightingSystem, TemperatureControl, \right. \\ \left. AccessControlSystem, MeteorologicalSystem, ElectricalSystem, time \right)$$

The algorithms use data from these systems and data sources that have direct or indirect impact on the target safety (S&S) and comfort (C&L) functions and on the minimization of energy consumption (REnC).

In reality, analyses need to be conducted separately for every building and specific algorithms to process data in the integrated building management system need to be devised.

4 Proposed Methodologies of Information Processing in Intelligent Buildings

Strong functional diversification of building automation subsystems implies a dissimilarity in information processing. There is often a need to use information possessed by other subsystems in order to optimize controls. A classic example of this is energy distribution and comfort control. The occurrence of such situations leads to the need for concurrent information processing and output sharing. To meet such objectives the identification of information processing methodologies

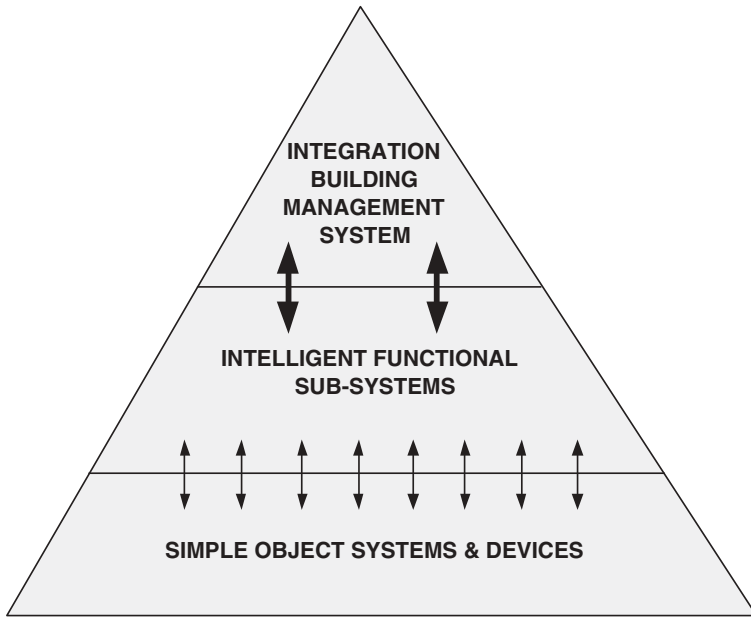


Fig. 2. Presents the principle of hierarchical integration from the lowest level of equipment and subsystems, through intelligent functional systems to the highest, supervisory level.

is required. A classic model consists in holding parallel computation processes in each of the subsystems. Distributed information processing where all subsystems are integrated can be realised by a two-layer model with a central processing unit. In reality, higher computational flexibility is ensured by a three-level model, where local clusters of integrated subsystems occur.

Allocation of computing power and the adaptation of algorithms to handle spots where the computing demand is the highest are interesting topics here. Building design and automation require the amount and types of information flow to be foreseen properly, so that functional, economic and environmental objectives are met by the adopted structure.

5 Analysis of Options for the Shaping of Technical Structure in Intelligent Buildings

The identification of the information processing methodology that is based on suitable algorithms and the division of tasks into appropriate computation levels requires the adoption of technical solutions. An assumption is made, like in industrial automation, that a distributed information processing system consists of so called intelligent (or capable of independent data processing) subsystems linked to each other via communications which ensure the transfer of information. Tasks

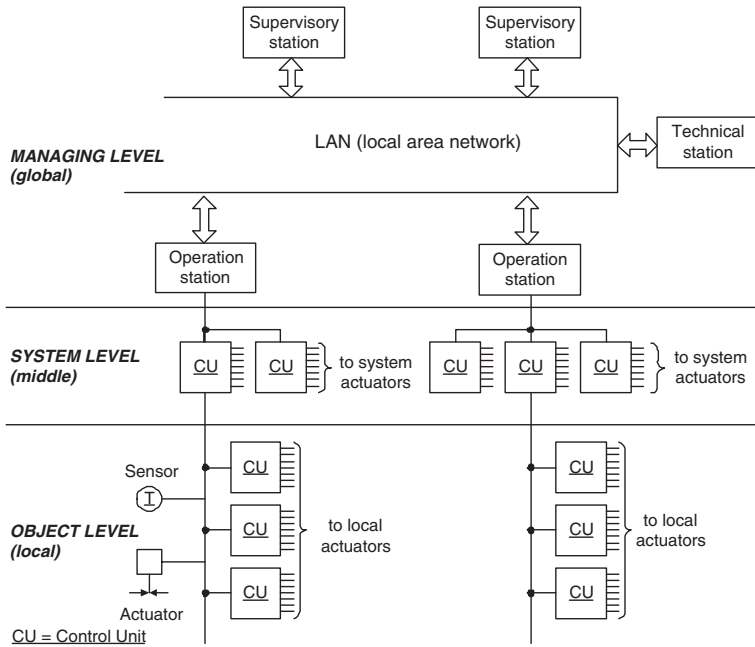


Fig. 3. Shows a general structure of the integration in the intelligent building.

to be performed by intelligent subsystems result from the subsystem's functions within the building automation structure. Communications links among system elements require transmission standards to be defined and the most appropriate communications protocols to be selected. In practice, there may also be the need to conform to various interfaces or subsystems from various manufacturers. One of the solutions frequently used for data acquisition from non-standard systems is the application of freely programmable controllers for data acquisition and control-signal generation. The system also uses intermediate or supervisory processing stations, whose major objective is to integrate subsystems and visualize and archive information.

The application of microprocessor-based controllers leads to hierarchical configurations in the IBMS systems [7]. The tasks of integration are performed at three levels: the local (object), intermediate (system) and global (supervisory) ones. At each of the levels there are devices with diversified information processing capabilities. The levels are linked (integrated) with each other in different ways. Local interfaces are used at the lowest level. Local field-buses connect devices at the intermediate level. An IT (LAN) network links process stations at the highest, supervisory level. The scale and complexity of the central control and management system in the building is always determined by the purpose the building serves.

The object (local) level contains sensors, executive devices and simple special-purpose controllers equipped with local dedicated software to exercise control and monitor at the object level. Such controllers include, for example, DDC (Direct Digital Control) controllers of process equipment (air-conditioning convectors, lighting, etc.), electricity or heat meters and smart servo-motors and sensors. In safety systems there are card readers or bio-metric readers of the access control system; active movement sensors of the burglary and break-in alarm system; carbon monoxide detection sensors, loop monitoring and control modules of the fire detection and alarm system. The communications bus forms a network of controllers at this level, thus making check points available to higher-level elements.

Within this layer the integration can take place at

- The software level (via a common data transmission protocol),
- The gateway level.

The software integration relates to the controllers that communicate with one another via an independent data transmission bus without using higher-level controllers or IBMS software for information exchange. Such integration can be achieved by the application of apparatus from one manufacturer (a single communications standard) or by the application of products from many manufacturers supporting one of open communications busses (such as LonWorks or BACnet [6]). The integration ensures that any network parameters of one piece of equipment can be accessed or modified by all devices that are linked into a network.

If products from various manufacturers non-supporting one of open communications busses are used, the integration can be achieved via hardware interfaces - gateways. The gateway is a device that provides access to physical and software points in devices based on a different communications standard, if the devices communicate via the system data bus.

The system (intermediate) level is usually built of system controllers. The system controllers are generally designed to control major system elements. They are, for example, air-conditioning controllers or lighting. Central processing units of the fire alarm, burglary and break-in alarm and closed-circuit television systems also fall into this category. They are able to communicate directly with lower-layer sensors and executive devices or indirectly via communications links with object controllers. The system controllers have most frequently built-in ports in order to link the controllers with a computer and initialize and set initial parameters. They can be programmed by higher-level devices. If the connection with the system breaks, the system controllers can continue their normal operation, maintaining full control of on-going operations.

Within this layer the integration can take place at

- The interface level,
- The hardware level.

The interface-based integration, like the object controller layer, facilitates the connection of specialized devices to the consistent IBMS system. Unlike the

previous layer, the integration via interfaces (gateways) makes it possible to add entire system buses from various manufactures or of different standards. With the application of gateway-type devices, one can integrate open communications busses (e.g. LonWorks, BACnet) with any other buses. They can be industrial standard buses M-Bus, ModBus (J-Bus), S-Bus or the manufacturers proprietary communications standards as for example, Sabroe, York ISN, Hiross, Grundfos, Wilo, Siemens, ABB.

The second type of integration within the system-controller layer is achieved by linking software-dedicated physical inputs with controller or active device outputs. Communications capabilities are limited here by the number of dedicated inputs and outputs.

Such integration can be applied to buildings, where the controllers require only supervision or, to a small extent, to be switched on by software, because they are provided with autonomous functions and sophisticated technical solutions.

The supervisory (global) level contains supervisory and working operator's stations. The operator's stations allow the technical staff to exercise holistic control of the BMS. They are most frequently PCs with essential software applications. In this layer, devices are typically integrated via a local computer network and network facilities (e.g. DDE, ODBC, OLE, etc.) for making the data base contents available.

Supervisory stations are at the top of the hierarchy and supervise all subsystems linked into the systems. This layer's operator has access to any information about every system element and can exercise control of it. In the case of alarm or emergency, a device at the supervisory layer can exercise control of the entire system by taking over the rights that are attributable to the lower layer. The supervisory layer devices are mainly responsible for collecting, storing and processing data and for generating reports that form the basis for conducting a long-term analysis and supporting intelligent building management.

6 The Role and Significance of the Integration of Building Automation Systems

The building automation issues presented so far indicate the need for communication connections and a holistic approach to information management. Functional subsystems need to be integrated into a single system in order to ensure the efficient optimisation of energy distribution, safety and comfort to building users and the effective operation of the system. Such a "digital nervous system" in a building, based on a supervisory computer, is called the Integrated Building Management System (IBMS). Concentration of all information on building-related events helps in the taking of proper decisions and the analysis of situations which are often complicated. The IBMS systems allow the combining of textual and numerical information with acoustic and visual data. This is of particular importance when a hazard to the building occurs. The possibility to re-configure resources is a very significant function of IBMS systems as the building

function management can be adjusted to changing users and user requirements. The linkage between information applications and the design documentation of the building is a natural feature of the system. Historical data on all events are archived and then used to conduct analyses and apply an expert methodology. In order to ensure full and sufficiently quick processing of information that is supplied to the IBMS, the system structure needs to be analysed during the system design phase. The best results may be obtained in distributed computation systems. The distribution of computation has the additional advantage of increased system reliability.

Consequently, the objective of the IBMS systems is to simplify and centralise monitoring, control and management of a single building or a number of buildings. Such systems are built to manage the building more efficiently, reduce labour and energy cost and create a user-friendly environment. The most important advantages of the IBMS systems are:

- simplification of the control of the system by programming procedures and repetitive operations,
- reduction of time needed to learn how to operate the system by the application of graphic interfaces,
- faster and more accurate reactions to individual needs and inconveniences of building users,
- reduction of energy costs by the introduction of central management and the implementation of energy management programs,
- management on the basis of historical records, system maintenance programs, automatic alerting in emergency situations,
- scalability and flexibility consisting of the ability to adjust to new requirements, organizational structures, sizes and growing requirements,
- more precise allocation of costs for individual users who use shared resources,
- improvement in the functionality of control through hardware and software integration of many subsystems, such as fire-alarm, access-control or lighting systems.

7 Examples of the Application of the IBMS in an Intelligent Building

Currently, every major building has qualities that are typical of the intelligent building. The greater its size, the more affluent the building is in electronic systems. The Control Tower at the Airport in Wroclaw is a good example of the application of the IBMS.

Below there is a list of all subsystems and devices integrated in the IBMS in the Airport Control Tower:

- A) A power supply system:
 - a) Electrical fittings in switch-gears: switches, stand-by activation system, lighting protectors,



Fig. 4. Shows the general appearance of this modern building (2001).

- b) Analyzers of power grid parameters in switch-gears,
- c) An UPS,
- d) A 100 kVA generating set,
- e) A battery for the emergency power supply of an individual emergency lighting system.
- B) Air-conditioning and ventilation systems:
 - a) Basic and stand-by controllers for inflation air-conditioning,
 - b) Basic and stand-by controllers for deflation air-conditioning,
 - c) A steam moistener,
 - d) Individual uptake ventilators in sanitary and service rooms,
 - e) Refrigerating units.
- C) Service systems in the Airport Control Tower:
 - a) A central heating system,
 - b) A warm-water supply system,
 - c) A hydrophore unit,
 - d) A cabling system of copper and fibre optic cables,
 - e) Temperature sensors in rooms where individual air-conditioning units are installed.
- D) Burglary and break-in alarm systems;
- E) A closed-circuit TV system;
- F) A fire-alarm system.

The systems integration in Airport buildings increases the safety of these buildings significantly. This is important for people working there and for air-traffic control equipment.

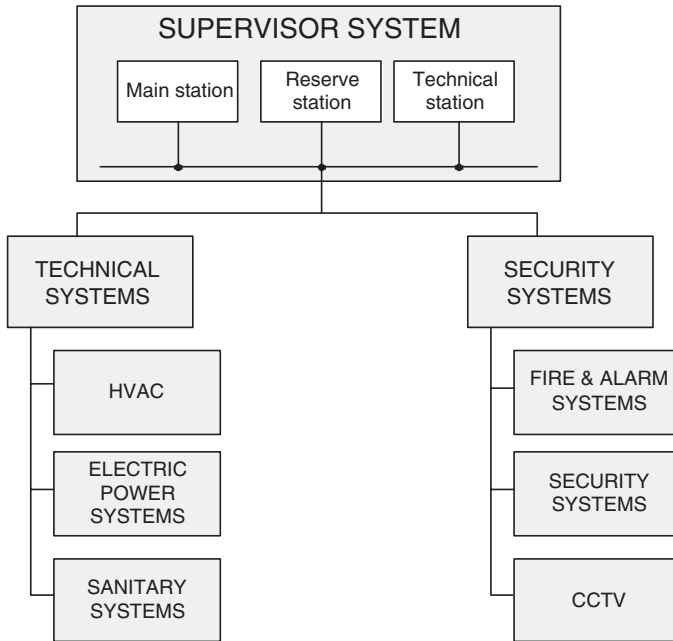


Fig. 5. contains a block diagram of the IBMS.

The integration also contributed to higher comfort of the staff and to the reduction of operational costs of the building. The efficient operation of the Flight Control Centre is a prerequisite for safe traffic in the airport. Owing to the application to the IBMS, the staff can interpret incoming data with the use of a transparent graphic interface, identify quickly sources of emergency situations and respond to emergency situations more efficiently. The IBMS is also used to archive data on any events that are reported to the operator's station as standard and to generate periodic reports on system operation.

8 Conclusions

This paper provides a review of the most important issues related to information processing in intelligent buildings. Hardware and software solutions adopted in The Wrocław Airport Control Tower have been presented. Integrated building management systems based on decentralized automation offer solutions to the most important questions underlying the functionality of intelligent buildings. Research objectives focus on processing distribution methods which would offer the independence of intelligent subsystems from each other, a possibility to optimize computations according to adopted quality criteria, access to all data at the proper time, a guarantee of the high reliability of the system as a whole and the minimization of outlays on technical infrastructure. Practical experience con-

firms that no contemporary building can be designed and then utilized without intelligent electronic systems integrated into the IBMS system. The operational costs of such buildings appear to be lower than those of traditional solutions. This means that searching for new and better solutions for the synthesis of the intelligent building makes profound economic and cognitive sense.

One thing can be taken beyond any doubt. The philosophy of the intelligent building is generally friendly towards man and the environment. We all have in common the aim of creating a friendly, supportive to work and efficient environment where an organization (firm) could attain its specified goals.

References

1. V.Boed, I.Goldschmidt, R.Hobbs,J.J.McGowan: *Networking and Integration of Facilities Automation Systems.*, CRC Press, 1999.
2. G. Huyberegts, P.M. Szczowka, B.W. Licznarski: *Gas sensor arrays for quantitative analysis and alarm generation*, Proc. IEEE International Symposium on Industrial Electronics, ISIE Guimaraes, Portugal, 7–11.07.1997, vol. 1 s. SS134–SS139.
3. B.W. Licznarski, P. Szczowka, A. Szczurek, K. Nitsch: *Non selective gas sensors and artificial neural networks – determination of gas mixtures.* In: Franz Pichler, Roberto Moreno-Diaz, Peter Kopacek (eds), Proc. 7th International Workshop Computer Aided Systems Theory, Vienna, 1999. Springer, Berlin, 2000, pp. 565–572, (Lecture Notes in Computer Science, ISSN 0302–9743; vol. 1798)
4. S.Sluszczak: *Integrated computer system for monitoring and management of specialized systems in an intelligent building.*, A master's thesis. Wroclaw University of Technology Faculty of Electronics, 2000.
5. C.Myers: *Intelligent Buildings.*, New York, 1996.
6. H.M.Hewman: *BACnet–Answer to Frequently Asked Questions.*, HPAC – Heating/Piping/Air Conditioning, No 3/3/1997, pp.47–51.
7. P.Wróbel: *Integration levels in Intelligent Building.*, IV International Intelligent Building, Wroclaw, 1999.
8. A.Czemplik, A.Jaboski,R.Klempous: *Properties of fieldbus networks vs. functionality of distributed control systems.*, EUROCAST 2001 – Formal methods and tools for computer science. [Eight International Conference on Computer Aided Systems – Theory and Technology]. Extended abstracts. Ed. by R. Moreno-Diaz and A. Quesada-Arencibia. Las Palmas de Gran Canaria, Spain, 13–19 February 2001. Las Palmas de Gran Canaria: IUCTC Universidad de Las Palmas de Gran Canaria, cop. 2001 pp. 130–135.
9. Microtech International Ltd: *Integrated Building Management System in the Airport Control Tower – Technical Documentation.* , Wroclaw, 2001.

Temporal Approaches in Data Mining. A Case Study in Agricultural Environment

Francisco Guil¹, Alfonso Bosch¹, Samuel Túnez¹, and Roque Marín²

¹ Department of Languages and Computation

University of Almería

04120 Almería (Spain)

{fguil,abosch,stunez}@ual.es

² Computer Science School

University of Murcia

30071 Espinardo (Murcia, Spain)

roque@dif.um.es

Abstract. In this work we present an study of the application of data mining techniques in an interesting domain, the Integrated Control. We found that in this dynamic domain, non-temporal data mining techniques are not suitable, needing to find a good temporal model. After a comparative study of several candidate models, we propose the use of the inter-transaction approach.

1 Introduction

Data mining is an essential step in the process of knowledge discovery in databases that consists of applying data analysis and discovery algorithms that produce a particular enumeration of structures over the data [9]. There are two types of structures: models and patterns. So, we can talk about local and global methods in data mining [21]. In the case of local methods, the simplest case of pattern discovery is finding *association rules* [1]. The initial motivation for association rules was to aid in the analysis of large transactional databases. The discovery of association rules can potentially aid decision making within organizations. Another approach is integrating the data mining process into the development of a Knowledge Based System [22]. The discovered association rules will serve us for validating the rules used by the expert system and for discovering new rules which can enrich the knowledge base of the expert system.

Since the problem of mining association rules was introduced by *Agrawal*, a large amount of work has been done in several directions, including improvement of the *Apriori* algorithm [2,24], mining generalized, multi-level, or quantitative association rules [27,28], mining weighted association rules [31], fuzzy association rules mining [13], constraint-based rule mining [12], efficient long patterns mining [11,17,32], maintenance of the discovered association rules [8], etc. We want to point out the work in which a new type of association rules was introduced, the *inter-transaction association rules* [19].

Our goal is to find a temporal data mining model suitable in the application domain named Integrated Control. One of the main problems in this domain is the selection of the best product (biological or chemical) to control a determinate pest or disease which is damaging a crop. This model will be integrated into a decision-making system for integrated control advising.

The remainder of this paper is organized as follows. Section 2 presents an study of the main temporal data mining techniques; Section 3 presents the inter-transactional approach as a temporal data mining technique; Section 4 presents a case study in agricultural environment; and Section 5 summarizes the conclusions and presents the future work.

2 Temporal Data Mining

Temporal data mining is an important extension of data mining techniques. It can be defined as the activity of looking for interesting correlations or patterns in large sets of temporal data accumulated for other purposes [5]. It has the capability of mining activity, inferring associations of contextual and temporal proximity, some of which may also indicate a cause-effect association. This important kind of knowledge can be overlooked when the temporal component is ignored or treated as a simple numeric attribute [26].

Data mining is an interdisciplinary area which has received contributions from a lot of disciplines, mainly from databases, machine learning and statistic. In [34] we found a review of three books, each one written from a different perspective. Although each perspective make strong emphasis on different aspects of data mining (efficiency, effectiveness, and validity), only when we simultaneously take these three aspects into account we may get successful data mining results. However, in the case of temporal data mining techniques, the most influential area is the artificial intelligence because its work in temporal reasoning have guided the development of many of this techniques.

In non-temporal data mining techniques, there are usually two different tasks, the description of the characteristics of the database (or analysis of the data) and the prediction of the evolution of the population. However, in temporal data mining this distinction is less appropriate, because the evolution of the population is already incorporated in the temporal properties of the data being analyzed.

We can found in the literature a large quantity of temporal data mining techniques. We want to highlight some of the most representative ones. So, we can talk about sequential pattern mining [3], episodes in event sequences [20], temporal association rules mining [4,16,14], discovering calendar-based temporal association rules [15], patterns with multiple granularities mining [5], and cyclic association rules mining [23]. In the literature there is a lot of work related to time series analysis. In this case, time-stamped scalar values of an ordinal domain generate curves and reveal trends. However, this characteristic is not extensible to time sequences, so no trends can be defined. Therefore, in this work, the time series analysis is not contemplated as a temporal data mining technique.

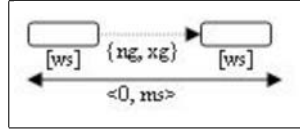


Fig. 1. Universal formulation of sequential pattern

In [10] M.V. Joshi et al. proposed an universal formulation of sequential pattern, which unifies and generalizes most of the previously proposed formulations as the generalized sequential patterns and episode discovery approach. They also introduced two new concepts. The first one is a directed acyclic graph representation of the structural and timing constraints of sequential pattern, and the second one is an approach that supplies several different ways for defining the support of a pattern, depending on the user's perception.

They assumed that the input is a data sequence characterized by three columns: object, timestamp, and events. Each row records occurrences of events on an object at a particular time.

The universal formulation of sequential patterns includes four timing constraints that can be used in the discovery process. In Figure 1 we can see the graphical representation of the universal sequential pattern. The meaning of each timing constraints is:

- ms (**Maximum Span**): The maximum allowed time difference between the latest and earliest occurrences of events in the entire sequence.
- ws (**Event-set Window Size**): The maximum allowed time difference between the latest and earliest occurrences of events in any event set.
- xg (**Maximum Gap**): The maximum allowed time difference between the latest occurrence of an event in an event set and the earliest occurrence of an event in its immediately preceding event set.
- ng (**Minimum Gap**): The minimum required time difference between the earliest occurrence of an event in an event set and the latest occurrence of an event occurrence of an event in its immediately preceding event set.

The mining process consists in discovering all sequential patterns which satisfy the support and timing constraints defined by the user.

In the next section we will talk about inter-transaction association rules. The one-dimensional inter-transaction association rules can be viewed as temporal association rules whenever the only dimension is the time.

3 Inter-transaction Association Rules

In [19], Hongjun Lu et al. introduced a new type of association rule, the *inter-transaction association rule*, extending the semantic of mining classical or intra-transaction association rules. The extension is focused on the discovering of associations among items within different transactions or data records. In addition, the inter-transaction approach deals with data records having dimensional attributes.

Table 1. A database with four transactions

T_{id}	Date	Items
t_1	$date_1$	a, b, c
t_2	$date_2$	a, b, c, d, e
t_3	$date_3$	a, f, g
t_4	$date_4$	e, h

Table 2. An extended database

Extended transactions	Extended items
t_1	$\Delta_0(a), \Delta_0(b), \Delta_0(c)$
t_2	$\Delta_1(a), \Delta_1(b), \Delta_1(c), \Delta_1(d), \Delta_1(e)$
t_3	$\Delta_2(a), \Delta_2(f), \Delta_2(g)$
t_4	$\Delta_3(e), \Delta_3(h)$

This dimensional attributes are a subset of the attributes that compose the dataset and they represent an interesting context for the existence of the associations, context which is ignored in classical data mining. Typical dimensional attributes are time, distance, latitude, etc. The name of dimensional attributes is due to the fact that they form a multidimensional space, and the transactions or the data records can be viewed as point in this multidimensional space. This type of attributes is also treated in [25] in the discovering of multidimensional sequential pattern.

We will show the difference between intra and inter-transaction association mining in Example 1.

Example 1. Consider the database depicted in Table 1. The set of attributes are Transaction-id, a dimensional attribute (for example, the date in which the events occur) and a list of items. If we apply a classical data mining algorithm (for example, the Apriori algorithm [3]) over this set of data with a value of 50 percent for the minimum support, we obtain the next enumeration of intra-transaction pattern:

$$\{a\}, \{b\}, \{c\}, \{e\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}$$

Now we will see how the inter-transaction data mining schema works. First, the database is transformed into an extended database as shown in Table 2. After such transformation, each record in the extended database will contain a value representing the relative address from a reference point in the dimensional space, and a list of items. Next, the algorithm extracts the large extended itemsets (the inter-transaction patterns) and the next phase is the association rule generation. In this case, the set of patterns discovered is:

$$\{\Delta_0(a)\}, \{\Delta_0(b)\}, \{\Delta_0(c)\}, \{\Delta_0(e)\}, \{\Delta_0(a), \Delta_0(b)\}, \{\Delta_0(a), \Delta_0(c)\}, \\ \{\Delta_0(b), \Delta_0(c)\}, \{\Delta_0(a), \Delta_1(e)\}, \{\Delta_0(a), \Delta_0(b), \Delta_0(c)\}$$

We want to highlight the pattern $\{\Delta_0(a), \Delta_1(e)\}$ which can be interpreted as "it is probably that the event e occurs one temporal unit after the occurrence of

event a". In this case, the context is not ignored (it is explicit in the syntax of the pattern) and the result is a set of pattern/association rules with more expressiveness and more prediction capabilities.

The first algorithm designed to obtain this type of association rules was named *E-Apriori* [19,18] and is an extension of the *Apriori* algorithm. This algorithm was presented with an improved version and was named *EH-Apriori*. The improvement was the incorporation of the hash technique presented in [24] to filter out unnecessary candidate 2-itemsets. In [29,30], *A.K.H. Tung et al.* presented an algorithm with the same goal named *FITI* (*Fist-Intra, Then-Inter*), but in this case the algorithm is not an Apriori-based algorithm. It was designed specifically for discovering frequent inter-transaction itemsets.

However, the three cited algorithms have some limitations. In [33] the authors proposed a modified and extended version of the multidimensional inter-transaction association rules, introducing a more general, practicable and reasonable schema. The improved version proposed was focused on the the introduction of different types of dimensional attributes, the adoption of association constraint mode to formulate inter-transaction association rules, and the definition of new support and confidence formulas.

4 A Case Study in Agricultural Environment

4.1 The Integrated Control Domain

Our goal is to find a temporal model of data mining suitable in the *integrated control domain*. This is a sub-domain of the global agricultural domain that, in the best of our knowledge, the data mining techniques have never been applied. The main problem in this domain is the selection of the best chemical or biological product to apply in order to control a concrete pest or disease under certain circumstances.

The actuation of the technicians and growers is controlled by a mark of quality called *Integrated Production* which was created and carried it out according to the Spanish office of patents and trademarks. It is understood as an agricultural production system that uses natural production resources and mechanisms to the maximum. Integrated Production includes extensive regulating standards, as well as monitoring and inspection by the corresponding authorities. When a group of growers decides to adopt the Integrated Production quality standard, it must submit to discipline in growing implying intervention by technicians, marketing control and periodical reviews by the certifying companies to see that the standard is being complied with.

The entities that appears in this domain are the crop, the field or greenhouse, the plants, the parasites (pest and diseases) and the auxiliary fauna (useful fauna). It can be viewed as a system which is affected by external variables (climate, humidity, ...) and, in order to keep its balance, control measures may be applied which are especially respectful of the crop, the useful fauna and the environment.

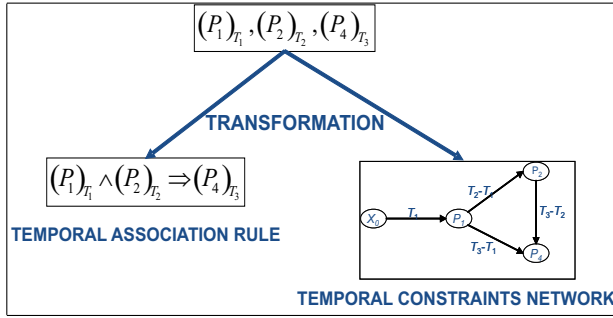


Fig. 2. Duality temporal association rule - temporal constraint

Biological and chemical product are carefully selected taking into account the requirements of society, profitability and environmental protection. Another key issue in the selecting of the right product is the temporal constraint amongst them. In other words, the chemical product can damage the biological products. So, the technicians must respect the persistence attribute of the biological products. This attribute indicates the number of temporal unit (days, weeks, etc.) in which several chemical product can not be apply during this interval of time. In addition, several incompatibilities amongst several chemical product exist. The simultaneous application of this products produces cross reaction causing damages to the crop. In this case, a temporal constraint must be satisfied too.

In the last three years, our group has been working in the development of a decision-making system for integrated control advising [7]. The core of this system is a Knowledge-Based System named *SAEPI*. This expert system makes a decision about the need of a treatment. The next step is the selection of the best treatment plan, based on a multicriteria decision model. This treatment consists usually of multiple actions, and these actions are affected by temporal constraints among them and with general cultivation scheduling. The next step is generate a schedule. In order to assess the suitability of the treatment plan, checking its consistency and generating automatically the schedule, a Disjunctive Fuzzy Temporal Constraint Networks model [6] has been integrated in the decision-making system.

Enhancing the system with an additional temporal data mining model will allow the extraction of new knowledge from the historical information stored in databases. This knowledge can be used to validate and complement the knowledge supplied by the expert. It will also allow the extraction of temporal constraints for treatment plans, and the prediction of future actions. The extraction of temporal constraints is interesting, because this type of knowledge is particularly difficult to handle by the domain experts. At most, they can give a narrative description of the problem. All this is due to the duality property of the discovered patterns, which can be transformed both into temporal association rules and into temporal constraints, as shown in Figure 2.

Table 3. Columns that compose the dataset

Date	API	Area	Species	Cultivars	Pest/Disease	Applied product
...	...	...	...	...	...	...

Table 4. Description of the dataset

API	Area	Species	Cultivars	Pests/Diseases	Applied products
1	31	Pepper	9	13	73
2	74	Grape	5	14	92
3	42	Grape	5	14	82
4	89	Grape	6	15	84
5	32	Grape	5	18	137
6	40	Grape	7	17	183

4.2 The Integrated Control Database

We want to apply data mining techniques in a database that describes the relationship between applied chemical and biological products and disease-pests. The global database contains more than 20 tables with a great number of numerical, categorical, and time attributes.

From the global database, we selected a set of data whose structure is the represented in Table 3. The Table 4 shows a description of the dataset. Our initial intention was the selection of a data mining model suitable in our application domain. After several experiences, we found that non temporal data mining techniques had several limitations and no useful knowledge can be discovered. The algorithm we used is a version of the apriori algorithm [3], which discover a set of classical association rules. Before starting with the experiments, we made three temporal abstractions on the data set, grouping the data and sorted by day, month and year. Figure 3 shows the number of transactions versus temporal unit, and Figures 4, 5, and 6 shows the number of rules versus minimum support for each temporal unit.

A common characteristic we found is the high confidence level and the little support in the discovered rules. The problem is due to the high ratio of change of the applied products. This change is mainly due to the presence of biological variables.

A problem with similar characteristics is studied in [4], where the authors proposed an extension of the apriori algorithm to discover temporal association rules. This approach presents the same limitations involved with the rest of algorithms that discover classical association rules.

After these results, we reviewed the most representative temporal models in data mining, selecting the inter-transactional schema, due to the reasons cited in previous sections.



Fig. 3. Number of transactions versus temporal unit

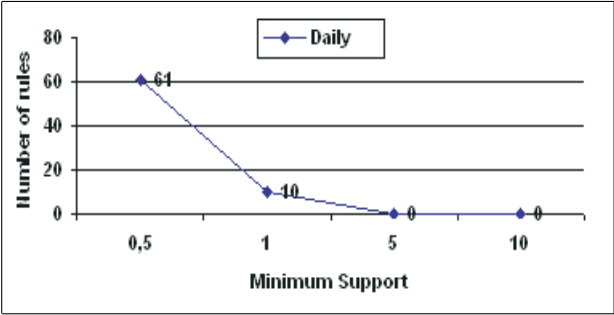


Fig. 4. Number of rules versus minimum support for the temporal unit *day*

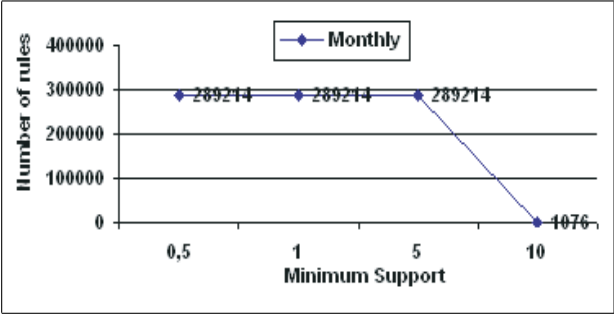


Fig. 5. Number of rules versus minimum support for the temporal unit *month*

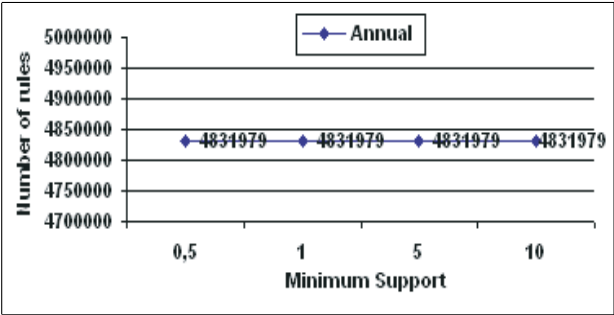


Fig. 6. Number of rules versus minimum support for the temporal unit *year*

5 Conclusions and Future Work

We have presented an integrated control database belonging to a dynamical domain. After several experiments, we obtained results that pointed out the need of using a temporal data mining model in order to obtain useful knowledge. We selected the one-dimensional inter-transaction model due to the nature of the temporal pattern (frequent itemsets) that can be discovered. In this case, the patterns can be translated into temporal constraints, and we can obtain temporal association rules from the discovered pattern. This type of association rule increases expressiveness and prediction capabilities.

As future work, we propose the study of a formal framework to specify the syntax and semantics of the temporal pattern/temporal association rules, based on constraint logic, temporal logic and fuzzy logic. The next step is the design and implementation of a temporal data mining algorithm taking into account the integration with a relational database management system. And finally, we propose to study the use of temporal constraint networks as post-data mining technique.

References

1. R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proc. of the ACM SIGMOD Int. Conf. on Management of Data, Washington, D.C., May 26–28, 1993*, pages 207–216. ACM Press, 1993.
2. R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In J. B. Bocca M. Jarke and C. Zaniolo, editors, *Proc. of 20th Int. Conf. on Very Large Data Bases (VLDB'94), September 12–15, 1994, Santiago de Chile, Chile*, pages 487–499. Morgan Kaufmann, 1994.
3. R. Agrawal and R. Srikant. Mining sequential patterns. In P. S. Yu and A. L. P. Chen, editors, *Proc. of the 11th Int. Conf. on Data Engineering, March 6–10, 1995, Taipei, Taiwan*, pages 3–14. IEEE Computer Society, 1995.
4. Juan M. Ale and G. H. Rossi. An approach to discovering temporal association rules. In *Proc. of the ACM Symposium on Applied Computing, Villa Olmo, Italy, March 19–21, 2000*, pages 294–300. ACM, 2000.
5. C. Bettini, X. S. Wang, and S. Jajodia. Testing complex temporal relationships involving multiple granularities and its application to data mining. In *Proc. of the 15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3–5, 1996, Montreal, Canada*, pages 68–78. ACM Press, 1996.
6. A. Bosch, M. Torres, and R. Marín. Reasoning with disjunctive fuzzy temporal constraint networks. In *Proc. of the 9th Int. Symposium on Temporal Representation and Reasoning (TIME'02)*, pages 36–43. IEEE Computer Society, 2002.
7. J. J. Cañadas, I. M. del Águila, A. Bosch, and S. Túnez. An intelligent system for therapy control in a distributed organization. In H. Shafazand and A. M. Tjoa, editors, *Proc. of the First EurAsian Conference on Information and Communication Technology (EurAsia-ICT'02), Shiraz, Iran, October 29–31, 2002*, volume 2510 of *Lecture Notes in Computer Science*, pages 19–26. Springer, 2002.

8. D. W. Cheung, J. Han, V. Ng, and C. Y. Wong. Maintenance of discovered association rules in large databases: An incremental updating technique. In Stanley Y. W. Su, editor, *Proc. of the 12th Int. Conf. on Data Engineering, February 26 – March 1, 1996, New Orleans, Louisiana*, pages 106–114. IEEE Computer Society, 1996.
9. U. Fayyad, G. Piatetky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AIMagazine*, 17(3)(3):37–54, 1996.
10. M. V. Joshi, G. Karypis, and V. Kumar. A universal formulation of sequential patterns. In *Proc. of the KDD'2001 Workshop on Temporal Data Mining, 7th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining San Fransisco, August 2001*. ACM, 2001.
11. R. J. Bayardo Jr. Efficiently mining long patterns from databases. In L. M. Haas and A. Tiwary, editors, *Proc. of the ACM SIGMOD Int. Conf. on Management of Data (SIGMOD'98), June 2–4, 1998, Seattle, Washington, USA*, pages 85–93. ACM Press, 1998.
12. R. J. Bayardo Jr., R. Agrawal, and D. Gunopulos. Constraint-based rule mining in large, dense databases. In *Proc. of the 15th Int. Conf. on Data Engineering, 23–26 March 1999, Sydney, Australia*, pages 188–197. IEEE Computer Society, 1999.
13. C. M. Kuok, A. W. C. Fu, and M. H. Wong. Mining fuzzy association rules in databases. *SIGMOD Record*, 27(1):41–46, 1998.
14. J. W. Lee, Y. J. Lee, H. K. Kim, B. H. Hwang, and K. H. Ryu. Discovering temporal relation rules mining from interval data. In *Proc. of the 1st EurAsian Conf. on Information and Communication Technology (Eurasia-ICT 2002), Shiraz, Iran, October 29–31, 2002*, volume 2510 of *Lecture Notes in Computer Science*, pages 57–66. Springer, 2002.
15. Y. Li, P. Ning, X. S. Wang, and S. Jajodia. Discovering calendar-based temporal association rules. *Data & Knowledge Engineering*, 44:193–218, 2003.
16. C. H. Lee, C. R. Lin, and M. S. Chen. On mining general temporal association rules in a publication database. In N. Cercone, T. Y. Lin, and X. Wu, editors, *Proc. of the 2001 IEEE Int. Conf. on Data Mining, 29 November – 2 December 2001, San Jose, California, USA*, pages 337–344. IEEE Computer Society, 2001.
17. D. Lin and Z. M. Kedem. Pincer-search: An efficient algorithm for discovering the maximum frequent set. *IEEE Transactions on Knowledge and Data Engineering*, 14(3):553–566, 2002.
18. H. Lu, L. Feng, , and J. Han. Beyond intra-transaction association analysis: Mining multi-dimensional inter-transaction association rules. *ACM Transactions on Information Systems (TOIS)*, 18(4):423–454, 2000.
19. H. Lu, J. Han, and L. Feng. Stock movement and n-dimensional inter-transaction association rules. In *Proc. of the Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD'98), Seattle, Washington, June 1998*, pages 12:1–12:7, 1998.
20. H. Mannila, H. Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.
21. Heikki Mannila. Local and global methods in data mining: Basic techniques and open problems. In P. Widmayer, F. Triguero, R. Morales, M. Hennessey, S. Eidenbenz, and R. Conejo, editors, *In Proc. of the 29th Int. Colloquium on Automata, Languages and Programming (ICALP 2002), Malaga, Spain, July 8–13, 2002*, volume 2380 of *Lecture Notes in Computer Science*, pages 57–68. Springer, 2002.

22. C. Ordóñez, C. A. Santana, and L. de Braal. Discovering interesting association rules in medical data. In D. Gunopulos and R. Rastogi, editors, *Proc. of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, Dallas, Texas, USA, May 14, 2000*, pages 78–85, 2000.
23. B. Özden, S. Ramaswamy, and A. Silberschatz. Cyclic association rules. In *Proc. of the 14th Int. Conf. on Data Engineering, February 23–27, 1998, Orlando, Florida, USA*, pages 412–421. IEEE Computer Society, 1998.
24. J. S. Park, M. S. Chen, and P. S. Yu. An effective hash based algorithm for mining association rules. In M. J. Carey and D. A. Schneider, editors, *Proc. of the 1995 ACM SIGMOD Int. Conf. on Management of Data, San Jose, California, May 22–25, 1995*, pages 175–186. ACM, 1995.
25. H. Pinto, J. Han, J. Pei, k. Wang, Q. Chen, and U. Dayal. Multi-dimensional sequential pattern mining. In *Proc. of the ACM CIKM Int. Conf. on Information and Knowledge Management, Atlanta, Georgia, USA, November 5–10, 2001*, pages 81–88. ACM, 2001.
26. J. F. Roddick and M. Spiliopoulou. A survey of temporal knowledge discovery paradigms and methods. *IEEE Transactions on Knowledge and Data Engineering*, 14(4):750–767, 2002.
27. R. Srikant and R. Agrawal. Mining generalized association rules. In U. Dayal, P. M. D. Gray, and S. Nishio, editors, *Proc. of 21th Int. Conf. on Very Large Data Bases (VLDB'95), September 11–15, 1995, Zurich, Switzerland*, pages 407–419. Morgan Kaufmann, 1995.
28. R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In H. V. Jagadish and I. S. Mumick, editors, *Proc. of the 1996 ACM SIGMOD Int. Conf. on Management of Data, Montreal, Quebec, Canada, June 4–6, 1996*, pages 1–12. ACM Press, 1996.
29. A. K. H. Tung, H. Lu, J. Han, and L. Feng. Breaking the barrier of transactions: Mining inter-transaction association rules. In *Proc. of the 5th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, August 15–18, 1999, San Diego, CA, USA*, pages 297–301. ACM Press, 1999.
30. A. K. H. Tung, H. Lu, J. Han, and L. Feng. Efficient mining of intertransaction association rules. *IEEE Transactions on Knowledge and Data Engineering*, 15(1):43–56, 2003.
31. W. Wang, J. Yang, and P. S. Yu. Efficient mining of weighted association rules (war). In *Proc. of the sixth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, August 20–23, 2000, Boston, MA, USA*, pages 270–274. ACM, 2000.
32. D. L. Yang, C. T. Pan, and Y. C. Chung. An efficient hash-based method for discovering the maximal frequent set. In *Proc. of the 25th Int. Computer Software and Applications Conference (COMPSAC'01), 8–12 October 2001, Chicago, IL, USA*, pages 85–93. IEEE Computer Society, 2001.
33. A. Zhou, S. Zhou, J. Wen, and Z. Tian. An improved definition of multidimensional inter-transaction association rule. In *In Proc. of the 3rd Pacific-Asia Conference on Methodologies for Knowledge Discovery and Data Mining (PAKDD-99), Beijing, China, April 26–28, 1999*, volume 1574 of *Lecture Notes in Computer Science*, pages 104–108. Springer, 1999.
34. Zhi-Hua Zhou. Three perspectives of data mining (book review). *Artificial Intelligence*, 143:139–146, 2003.

Personalized Guided Routes in an Adaptive Evolutionary Hypermedia System

Nuria Medina-Medina¹, Fernando Molina-Ortiz¹, Lina García-Cabrera²,
and José Parets-Llorca¹

¹Depto. Lenguajes y Sistemas Informáticos
University of Granada (SPAIN)
{nmedina, fmo, jparets}@ugr.es

²Depto. Informática
University of Jaén (SPAIN)
lina@ujaen.es

Abstract. In this paper we describe an adaptation method for adaptive hypermedia systems, consisting in personalized guided routes for the SEM-HP model. SEM-HP is a layered, systemic, semantic and evolutionary model for the development of adaptive hypermedia systems, which adapt to the particular features and interests of each user. For evolution it uses a Metasystem, which offers to the author a set of evolutionary actions that permit the hypermedia system to evolve in a flexible and consistent way. In SEM-HP a hypermedia system is composed by four subsystems, each of which offers a different functionality to the user and to other subsystems. User adaptation is carried out by the learning subsystem, which keeps and updates an user model, which includes the user knowledge about the informational elements offered by the system, his preferences and his goal, which is to reach a certain degree of knowledge. Guided routes direct the user through the hypermedia system, so the user goal can be reached in an optimal way.

1 Introduction

Hypermedia systems have been widely used to communicate knowledge, specially in the last years due to the success of the World Wide Web. Shnedierman [9] defines hypermedia as “a database that has active cross-references and allows the reader to “jump” to other parts of the database as desired”. Hypermedia systems have two elements: nodes and links. The nodes are the information units offered by the system, and can be text, images, audio, video, etc. The links connect the nodes, and allow the user to navigate through the system, thereby allowing the material to be viewed in a non-linear order. The structure of a hypermedia system can be complex, and the user can have problems of comprehension and disorientation, due to the large amount of information provided by the hypermedia system. Therefore, the main problem is that the user does not feel comfortable using the hypermedia system. The solution is to adapt both the links structure and the information to the features and interests of each user, which is performed by Adaptive Hypermedia Systems (AHS). In order to carry out the adaptation, the AHS must know the user. This is done by means of a user

model, which is created by the AHS and updated while the user navigates. The user model is the internal representation that the system has of the user, and it stores the knowledge, interests, goals, preferences, experience and personal data of the user. As we can see in [1], for Brusilovsky an AHS must satisfy three criteria: it must be a hypermedia system, it must have an user model, and it must be able to adapt the hypermedia system using the user model. In this paper, firstly we describe the SEM-HP model, then we detail the learning subsystem, which includes the user model, the weight, knowledge and update rules, and the adaptive methods. After that, we describe in more depth the guided route adaptive method. Finally, conclusions and further work are exposed.

2 SEM-HP Model

SEM-HP [4,5] is a Semantic, Systemic and Evolutionary model for the development of AHS, which therefore pretends to ease the navigation of the users, and also the task of design done by the author. SEM-HP model proposes:

- A software engineering process for the development of AHSs.
- A systemic and evolutionary architecture.
- An author tool, which permits the creation and evolution of AHSs following the development process.

The *development process* in SEM-HP is divided in four phases, which are iterative and evolutionary, that is, each phase integrates the changes performed by the developer in a flexible and consistent way. The phases (Figure 1) are the following:

- Conceptual phase. The author constructs conceptual and information worlds.
- Presentation phase. The author selects different presentations of a concrete conceptual and information world.
- Navigation phase. The author states how the user can browse the offered information.
- Learning phase. The author resolves the aspects of adaptation. Here, he answers the four essential questions in user adaptation: What? Whom? How? and When to adapt?

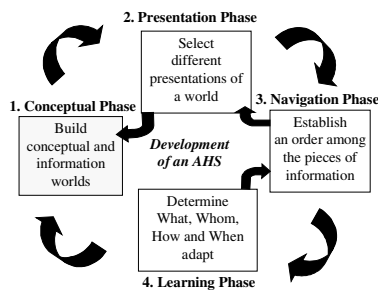


Fig. 1. Development process in SEM-HP

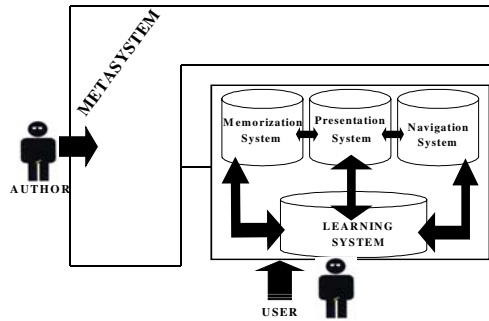


Fig. 2. Structure of SEM-HP model

The proposed *architecture* (Figure 2) performs a double division, horizontal and vertical. The horizontal division considers two levels of abstraction: System and Metasystem. The System is used by the user in order to navigate and read information, and the Metasystem is used by the author in order to develop and change the AHS. The vertical division structures the system in four subsystems. Each subsystem captures some features of the model and consequently it offers certain functions to the user and to the other subsystems.

Since AHSs offer the evolving knowledge captured by their authors, they need to change frequently. To support these changes, the Metasystem provides to the author a set of evolutionary actions. An evolutionary action is only executed if it satisfies a set of restrictions, which preserve the consistency of the system. The author can modify some restrictions using a special set of evolutionary actions, whose restrictions are metarestrictions. In addition, the metasystem also supports internal and external propagation of changes, that is, when the author performs a change in a subsystem, it is propagated in the same subsystem (internal), and to others (external), so consistency inside each subsystem and between them is guaranteed.

In the vertical division, SEM-HP conceives an AHS as composed by four interrelated subsystems: memorization, presentation, navigation and learning.

The *memorization subsystem* is in charge of storing, structuring and maintaining the knowledge offered by the system. The main element is the conceptual structure (CS), which is a semantic net with two types of nodes: concepts and items. Concepts are ideas labelled semantically, and items are the pieces of information offered by the system. The concepts are related by means of conceptual associations, and concepts and items are associated through functional relations. An item can have several attributes, such as author, date of creation, date of edition and type of media (which can be text, video, image, hypermedia, etc). In addition, every item has a role within each concept, which can be introduction, definition, example, bibliography, algorithm, comparison, analysis, extension, etc [3]. This role is represented in the functional association. We can see an example of conceptual structure in Figure 3.

The *presentation subsystem* allows the author to prepare different views of the same information, creating personalized CSs. For doing this, he selects different subsets of the original CS. For example, the author can create the presentation shown in Figure 4 from the conceptual structure of memorization in Figure 3.

The *navigation subsystem* allows the author to state the navigability of the conceptual relations. A conceptual relation can only be followed in the direction of the arrow, but if desired the author can extend its navigability in the other direction.

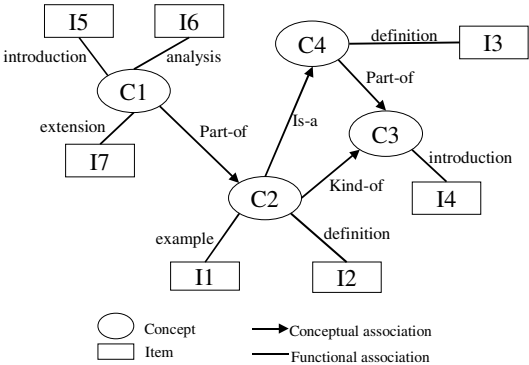


Fig. 3. Example of Conceptual Structure

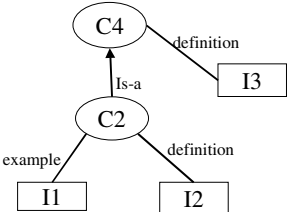


Fig. 4. Example of a presentation CS

The *learning subsystem* is in charge of performing the user adaptation, as we will show in the following section.

3 Learning Subsystem

This subsystem [7,8] is in charge of adjusting the navigation process to the features and interests of the current user. Its main elements are the user model, the adaptive methods and the weight, update and knowledge rules.

3.1 User Model

The user model stores and updates the information about the user which will be taken into account by the system to perform the user adaptation. The user model manages:

- The degree of knowledge the user has about every node in the CS.
- The user goal.
- Information about the preferences of the user.
- The user interests (the items whose visits lead to the goal).
- Personal information.

Table 1. Semantic labels for user knowledge

Label	Value
Null	1
Very Low	2
Low	3
Medium	4
High	5
Very High	6
Total	7

Table 2. User preferences

	Attribute	Values
Item information	Author	a string with the author name
	Date of creation	date*
	Date of edition	date*
	Type of media	text, image, audio, video, animation, execution, hypermedia, etc
Role	Role of the item	introduction, definition, example, bibliography, algorithm, comparison, analysis, extension, etc.
Navigation	Guided route length	Shortest

* A date can be: specific date, dates after a certain date (>, >=), dates before a certain date (<, <=) or a date interval (between)

The *degree of user knowledge* about the concepts and items in the CS is represented by a semantic label. There are seven different labels, as shown in Table 1. Each label has associated an equivalent numerical value. The degree of user knowledge is formally represented by a knowledge state:

$$K_{state} = \{ (n_1, label_{SEM}^1), (n_2, label_{SEM}^2), \dots, (n_N, label_{SEM}^N) \} \quad (1)$$

where $n_1, \dots, n_N$ are the nodes in the CS, and $label_{SEM}^i$ are semantic labels (Table 1)

The *user goal* will be to reach a certain amount of knowledge about some nodes in the CS. It is expressed as in (1), but it does not have to include all the nodes in the CS.

The *preferences* are information about some features of the items or of the navigation that the user prefers. Examples of these could be “I prefer items written by the author Joe Smith”, “I prefer documents which are not images”, or “If I specify a knowledge objective, I would like the system to guide me through the shortest route to achieve it”. In addition, the user is allowed to specify which of these preferences are more important to him. The attributes that can be used to express the user preferences are those shown in Table 2. To specify his preferences, the user gives a significance weight to each attribute, and, for each attribute, he specifies the values he prefers and he does not prefer, ordered. Hence, for each attribute there are two lists, one with the values the user prefers (affirmative list), and another one with the values he wants to avoid (negative list). Values in a higher position of the list show more liking (or disliking, depending on the list) of the user for that value. The user can place more than one value in the same position of the list, indicating the same preference for both.

3.2 Weight, Update, and Knowledge Rules

The user model is updated during the user navigation. This is done through the weight and update rules, which calculate the user knowledge about the concepts and items in the CS. In addition, the user navigation is restricted in function of the knowledge the user has, by means of the knowledge rules.

The *weight rules* calculate the user knowledge about a concept from the user knowledge about the items that are associated to the concept. Each item i_j is weighted with the influence p_j that the knowledge about that item $K(c.i_j)$ has on the knowledge about the concept $K(c)$, so it is calculated as follows:

$$K(c) = \sum_j p_j * K(c.i_j) \quad (2)$$

For example, in the weight rule in (3), which refers to concept C1 in Figure 3, the user knowledge about the item I7 is more influential in the user knowledge about the concept C1.

$$K(C1) = 0.25 * K(I5) + 0.25 * K(I6) + 0.5 * K(I7) \quad (3)$$

The *update rules* update the user knowledge about the items. When the user visits one item, his knowledge about that item and possibly about other items is updated in the user model. The *left-hand side* of the rule is a logic predicate $Visit(c.i)$ which becomes true every time the user visits the item $c.i$, and the *right-hand side* is a set of predicates, in which each predicate updates the user knowledge about a different item, as shown in (4).

$$Visit(c.i) \rightarrow Update(c.i), Update(c.i_1) \dots Update(c.i_k) \quad (4)$$

An update can be:

- Incremental or fixed: The knowledge about the item can be increased, or it can be set to a fixed degree of knowledge.
- Absolute or relative: It can use an absolute semantic label (Table 1) or the degree of user knowledge about the item in the left-hand side.
- First-time or Each-time: The update can be run only the first time the item in the left-hand side is visited, or every time it is visited.

Each predicate in the right-hand side of the rule represents an update that can be of any of the types just specified. The predicates corresponding to each kind of update can be seen in the lower level of the diagram shown in Figure 5. Updates are each-time by default; when they are first-time the suffix *first* is used in the name.

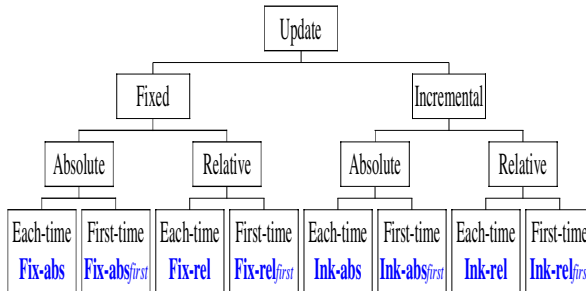


Fig. 5. Update predicates

Table 3. Example of update rules

Update rule	Action
Visit(I1) $\rightarrow$ Ink-abs(I1,3), Ink-rel(I3,1)	$K(I1)=K(I1)+3$, $K(I3)=K(I1)+1$
Visit(I2) $\rightarrow$ Fix-abs(I2,"total"), Ink-abs(I3,3)	$K(I2)=total$, $K(I3)=K(I3)+1$
Visit(I3) $\rightarrow$ Fix-abs(I3,"total")	$K(I3)=total$

Table 4. Restrictions in knowledge rules

	Strict	Not strict
Equality	$K(c.i) = l_{SEM}$	
Larger	$K(c.i) > label_{SEM}$	$K(c.i) \geq label_{SEM}$
Smaller	$K(c.i) < label_{SEM}$	$K(c.i) \leq label_{SEM}$
Interval	$label_{SEM,1} < K(c.i) < label_{SEM,2}$ Where $label_{SEM}$ are semantic labels and $label_{SEM,1} < label_{SEM,2}$	$label_{SEM,1} \leq K(c.i) \leq label_{SEM,2}$
OR_{SFT}	$K(c.i) \in [restriction_1, restriction_2, \dots, restriction_N]$ which means that $K(c.i)$ satisfies at least one restriction in the set	

For example, in Table 3 we show a possible set of update rules for the CS in Figure 4, and what they would do each time the item in the left-hand side is visited.

The *knowledge rules* state the knowledge restrictions the user must satisfy in order to visit each item. An order rule is associated to an item, and determines what items must be previously known by the user and what is the degree of knowledge needed in order to reach the item. They have the form shown in (5). The supported restrictions are shown in Table 4.

$$Restriction(c_i, i_1) op_L \dots op_L Restriction(c_k, i_k) \rightarrow Visitable(c.i) \quad (5)$$

where $Restriction(c_i, i_1) = (K(c_i, i_1) op_R \text{"label"})$, op_L is a logical operator and op_R is any operator included in Table 4.

An example of knowledge rule for item I3 in Figure 4 could be:

$$K(I1) \leq \text{"high"} \text{ or } K(I2) > \text{"medium"} \rightarrow Visitable(I3) \quad (6)$$

3.3 Adaptive Methods

The system applies adaptive methods in order to carry out the user adaptation, based on the information stored in the user model. The adaptive methods supported in SEM-HP are:

- Personalized views. For each user, the presentation CS that best fits his features and interests is selected.
- Orientation support. The user browses through the CS, so he has a global view of the link structure. Visited items are marked in a special color. In addition, the user knowledge about items and concepts is visualized in the CS (showing next to each node a semantic label), so the user can check how his knowledge increases while he browses the system.
- Navigation restricted by knowledge. All users are not ready to access and understand the same information, therefore the user navigation is restricted according to the knowledge rules defined by the author. As a result, items for

which a knowledge rule is not satisfied by the current knowledge state of the user are hidden and disabled.

- **Guidance.** With the aim of guiding the user as he navigates, SEM-HP supports two adaptive methods. On the one hand, desirable items are marked in a special color (desirable items are those that lead the user to his goal, and are accessible by the user). On the other hand, the system provides guided routes, which are discussed in the following section.

4 Guided Routes

Among other methods, SEM-HP uses personalized guided routes to ease the navigation of novice users. This method guides the user during his navigation with the aim of bringing him closer to his objectives and taking him apart from the information for which he is not prepared. A guided route is a set of items and an order to visit them. This route will take the user from its current knowledge state to his goal (that is, the desired knowledge state). To calculate it, the system uses a navigation tree, based on the knowledge and update rules, and on the preferences of the user. Firstly, we will show how the navigation tree is built, secondly, how a guided route is calculated and, finally, we will describe the changes that can be done in the navigation tree so the user preferences are taken into account.

4.1 Navigation Tree

In the navigation tree each node represents a knowledge state in which the user can be, and each arc represents the visit of an item. A parent node in the tree will have as many children as items the user can visit when having the knowledge represented by the parent node (that is, items whose knowledge rule is satisfied). Children nodes are obtained applying the update rules of those visitable items on the knowledge state represented in the parent node. In Figure 6 we can see the navigation tree generated from the CS in Figure 4 and the update and knowledge rules in Table 3 and Equation 7. The root node represents null knowledge about all items.

We observe that the navigation tree can be built only once for each CS (not every time a guided route is requested), and will only have to be rebuilt when the author changes the CS or the weight, update or knowledge rules. As described later, the arcs will be labelled to suit each user's preferences, but the knowledge and update rules will not have to be fired again.

4.2 Building Guided Routes

The route is obtained performing a search in the navigation tree. The starting node represents the knowledge state the user has, and the final node represents a knowledge state that matches the user goal. The arcs followed in the path from the start to the final node give the list of items that form the guided route. As seen, the knowledge goal of the user does not have to include all the items in the CS. Hence, when

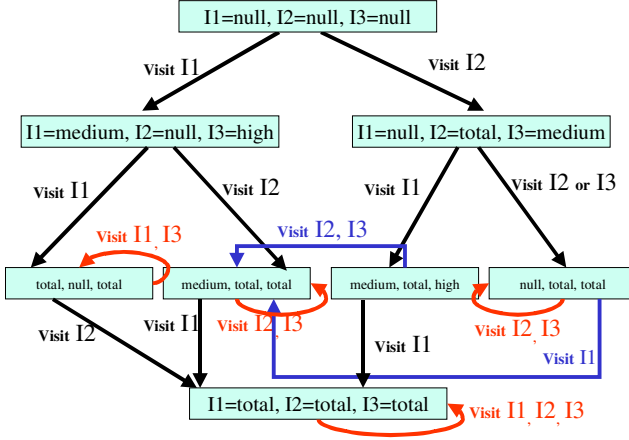


Fig. 6. Navigation tree

performing the search we will only take into account the knowledge about the items specified in the goal. That is, *CurrentState* can be a final node in the search when:

$$\forall (n_g, k_g) \in \text{Goal}, (n_c, k_c) \in \text{CurrentState}, n_g = n_c \Rightarrow k_c \geq k_g \quad (7)$$

4.3 Adapting the Navigation Tree to the User Preferences

We can build a guided route taking into account user preferences. Since a knowledge objective has been provided, we assume that a shorter rule is always preferred, being every thing else equal. This means that if there are two possible routes that lead to the knowledge objective, both of which go through items that are the same desirable for the user, we will choose the shortest one. Conversely, a longer route may be chosen if it matches sufficiently better the user preferences. To achieve this we label each arc in the tree with a number that summarizes the user preferences. A smaller number represents that following that arc (visiting the associated item) is more desirable regarding the preferences of the user. With this method, the chosen guided route will be the one that provides the shortest path in the tree, being the distance between two nodes the label of the arc just mentioned, and the starting and final nodes the ones that represent the present and desired knowledge states.

As mentioned, the label of the arc represents the user preferences. Being $A=\{a_1 \dots a_n\}$ the n attributes in the user preferences, $aw_1^o \dots aw_n^o$ the weights of the attributes, I the item whose visit the arc implies, and SGRL the shortest guided route length attribute, the label of each arc is calculated as follows:

1. The attribute weights are normalized so they sum one:

$$aw_i = \frac{aw_i^o}{\sum_{j=1}^n aw_j^o} \forall i = 1 \dots n \quad (8)$$

2. For each attribute, we calculate a weight for each value of each list. For both lists, they are calculated as follows:

$$vw_i^a = \frac{nEl - (i - 1)}{nEl} \quad (9)$$

where vw_i^a is the weight for the value in the i -th position of the list in attribute a , and nEl is the number of elements in that list. For example, a list with four values would have the weights 1, 0.75, 0.5 and 0.25.

3. We search the (attribute, list, value) triplets that are in the user preferences and are matched by any feature in the item I , except the SGRL attribute, that can not be calculated from the visit to I . We place in the set M the triplets found, and, if there is in M more than one value for the same attribute and the same list, we only keep the one with the highest weight, so in M there is only one value per attribute per list. We name $afw(a)$ the weight of the value that appears in the affirmative list of attribute ' a ' and in M , and $negw(a)$ the weight of the value that appears in the negative list of attribute ' a ' and in M .
4. The label of the arc r is:

$$l_r = 1 - \sum_{i=1}^{|A|} aw_i \cdot (afw(a_i) - negw(a_i)) \quad (10)$$

A label cannot be negative, since the value weights are calculated and M is constructed so afw and $negw$ are never larger than one, and the attribute weights are normalized. The term that subtracts from one in (10) becomes larger as the visit to I better matches the user preferences, so the label of the arc will become smaller. The SGRL preference is used in the normalization, but not in M . Due to the normalization of the attribute weights, if the SGRL attribute has a weight of w_s , the mentioned subtraction term in (10) can be at most $1 - w_s$. Hence, a higher SGRL preference value will make less important the other preferences (they subtract less), so it will make the sole number of steps more influential in the obtained guided route.

In Figure 7 we show an example of a simple navigation tree in which there are two possible routes to achieve the knowledge goal. For simplicity, we only take into account to attributes: language (with only Spanish in the affirmative list), and SGRL (shortest). Labels in trees (a) and (b) are calculated using different user preferences, which only vary in the attribute weights. In case (a), where language is more important, the longest route is chosen because the items in it are in the desired language. In case (b), where SGRL is more important, the algorithm chooses the route with less items.

5 Conclusions and Further Work

As defined, SEM-HP is a semantic, systemic and evolutionary architecture for the development of adaptive hypermedia systems. It is semantic because it supports a semantic net (the CS) that permits the author to represent and structure his knowledge and the user to navigate the information offered by the system. It is a systemic model because it conceives the AHS composed by four interrelated subsystems:

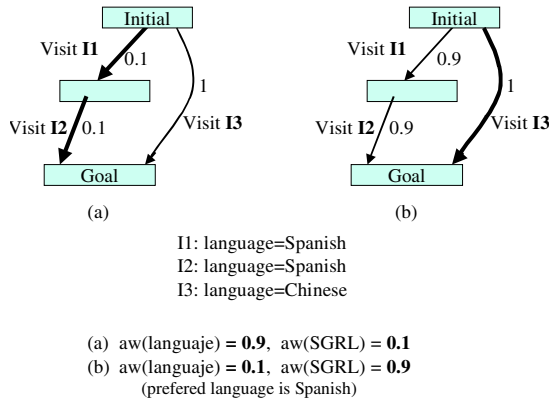


Fig. 7. Example of personalized guided routes

memorization, presentation, navigation and learning. SEM-HP is an evolutionary model because it has an architecture with two levels, in which the metasystem allows the author to flexibly perform changes in the AHS while the consistency is guaranteed. Finally, it is an adaptive model because it applies adaptive methods which permit adjusting both the links structure and information to the features and interests of each user captured in the user model.

Many different models, architectures and software engineering approaches have been proposed for the development of AHS. A good review can be seen in [6], but there are other approaches such as RICH [10], AHAM [2], etc. Although most adaptive hypermedia include a domain and an user model, in most cases the consistent evolution of the system is not explicitly considered, as in SEM-HP. Most revised models tend to focus in only one aspect of adaptation (to the user knowledge, to the user preferences, etc), or in few adaptive methods. SEM-HP tries to adapt at the same time to different features (for example, the personalized guided routes described here adapt both to the user knowledge and preferences), and uses a wide range of adaptive methods.

A tool based on the SEM-HP model is being developed, and now it implements the memorization, presentation and navigation subsystems. Our future work is centered on finishing the prototype, so we can validate the usefulness of the model in both the creation and evolution of hypermedia systems and the improvements in navigation due to user adaptation, including personalized guided routes.

References

1. Brusilowsky, P.: Methods and techniques of adaptive hypermedia. *User Modeling and User Adapted Interaction*, v 6, n 2–3. (1996) 87–129
2. De Bra, P., Houben, G., Wu, H.: AHAM, A Dexter-based Reference Model for Adaptive Hypermedia. *ACM Conference on Hypertext and Hypermedia*. Darmstadt, Germany. February 21–25, (1999)
3. García-Cabrera L.: SEM-HP: A Systemic, Evolutionary, Semantic Model for Hypermedia System Development (in Spanish). Ph Thesis. (2001)

4. García-Cabrera L., Rodríguez-Fórtiz, M^aJ., Parets-Llorca, J.: Formal Foundations for the Evolution of Hipermedia Systems. 5th European Conference on Software Maintenance and Reengineering. FFSE. Lisbon (2001)
5. García-Cabrera, L., Rodríguez-Fórtiz, M^aJ., Parets-Lorca, J.: Evolving Hypermedia Systems: a Layered Software Architecture. Journal of Software Maintenance and Evolution: Research and Practice, John Wiley & Sons, Ltd, 14 (5), (2002) 389–405
6. Lowe, D., Hall, W.: Hypermedia and the Web. An Engineering Approach. WileyEurope (1999)
7. Medina-Medina, N., García-Cabrera, L., Rodríguez-Fórtiz, M^aJ., Parets-Llorca, J.: Adaptation in an Evolutionary Hipermedia System: Using Semantic and Petri Nets. AH'2002. Málaga (2002)
8. Medina-Medina, N., García-Cabrera, L., Torres-Carbonell, JJ., Parets-Llorca, J.: Evolution in Adaptive Hipermedia Systems. IWPSE'01. Orlando (2002)
9. Shneiderman, B., Kearsley G.: Hypertext Hands-On!: An Introduction to a New Way of Organizing and Accessing Information. Addison Wesley (1989)
10. Wang, W., Rada, R.: Structured hypertext with domain semantics. ACM Transactions on Information Systems. 16(4) (1998) 372–412

Temporal Data Management and Knowledge Acquisition Issues in Medical Decision Support Systems^{*}

M. Campos, J. Palma, B. Llamas, A. González, M. Menárguez, and R. Marín

Artificial Intelligence and Knowledge Engineering Group
University of Murcia. Campus de Espinardo. Murcia 30071. Spain
`mcampos@dif.um.es`

Abstract. The development of data-intensive systems in medical domains has received increasing attention in recent years. In this work we present in depth some parts of ACUDES (Architecture for Intensive Care Units Decision Support) in which traditional techniques for managing and representing time have been integrated with a temporal diagnosis task in a decision support platform. ACUDES has been designed to manage the information regarding patient evolution and to describe the patients evolution in terms of the temporal sequence diseases suffered. These functionalities are supported by an ontology which simplifies the knowledge acquisition and sharing, while guaranteeing the semantic consistency of patients' evolution data. This work will be focused on the Temporal Data Management and the Knowledge Acquisition Tool.

Keywords: Decision Support, Temporal Information Management, Medical Knowledge Acquisition and Representation.

1 Introduction

In recent years, research in Artificial Intelligence in Medicine (AIM) has shifted from knowledge-intensive systems to data-intensive systems, and static consultation systems are followed by dynamic ones which capture better the patients evolution over time [1]. This new approach has given rise to an increasing interest in the development of decision support systems in which the temporal dimension plays an important role. The intense research in this area has also shown that the use of information and knowledge technologies is an important way of improving the quality of the services provided in Intensive Care Units (ICUs). However, and on account of the reactive component of tasks carried out in ICUs, diagnostic expert systems have been replaced by more efficient decision support systems and medical research tools. Another important conclusion is that the use of temporal representation, reasoning and abstraction techniques is essential, due to the intrinsic dynamic component in the ICU domain.

^{*} This work is supported by the Spanish MCYT under the project "Information Systems and Intelligent Agents in ICU", project number TIC2000-0873-C02-02.

It is in this context that we have designed ACUDES [2], a generic architecture for decision support systems for ICUs. ACUDES was built to deal with the heterogeneous nature of the information regarding the patient evolution. The information generated at ICUs can be related to either electro-hemodynamic signals or patient clinical history. Two main functionalities have been considered in ACUDES for decision support purposes: Firstly, ACUDES allows physicians to explore the complete patients' data, those from the clinical history (administrative data, test labs,...) and those from monitored signals. Furthermore, they can be considered within the temporal context (temporal relations with the other data) in which they are produced. Secondly, from the data collected, ACUDES can explain the patients evolution as a collection of diseases suffered, including their temporal and causal context (causal relations between diseases, and between diseases and their observations).

Therefore, the inclusion of modules which make temporal reasoning on patients' evolution data possible is essential. ACUDES temporal reasoning capabilities are provided by FuzzyTIME [3], a generic temporal reasoner which can be easily integrated into any application that requires management of temporal information, and which provides a high level language for performing temporal queries. Furthermore, a decision support systems for the ICU domain has to deal with a large amount of data provided by both monitored signals and clinical history data. In this context, the incorporation of such generic temporal reasoning in data base systems is essential. From the temporal reasoning perspective, temporal knowledge usually captured by a constraint network can be represented more effectively if the network is complemented by a database for storing the information typically associated to nodes of the network [4].

The rest of the paper will be focused on the Temporal Data Management and the Knowledge Adquisition Tool. The structure of the paper is as follows. In Section 2 the general architecture of ACUDES is presented. Then, the temporal data management subsystem is described in Section 3, where a brief introduction to the temporal reasoning module can be found. The structure of the data base as well as the different type of temporal information considered is put forward in section 3.1. In the same section, the integration of FuzzyTIME with the database is introduced, where special attention is laid on the kind of queries that can be solved by the system. The third part of the paper is focused on the description of the TBM and the knowledge acquisition tools designed for building the KB. In Section 4.1 the structure underlying the TBM is presented. A brief description of the ICU ontology is shown in Section 4.2. The architecture for the KA tools will be presented in Section 4.3. Finally, we provide some conclusions and future works.

2 General Architecture of ACUDES

The ACUDES architecture is shown in Figure 1. Exploration capabilities on patient evolution data requires a module for patient information management. Due to the importance of temporal dimension of patient evolution data in ICUs,

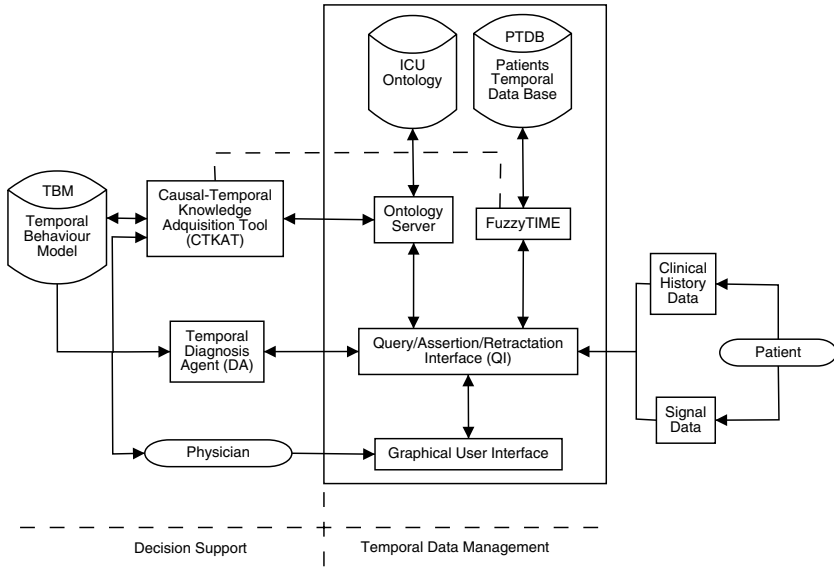


Fig. 1. ACUDES General Architecture

patients' data are handled by a temporal data base, the Patients Temporal Data Base (PTDB). Temporal reasoning capabilities are also necessary to guarantee the temporal consistency of inserted data and to infer new temporal relations from data inserted in PTDB, apart from those explicitly defined by users. This is very useful for decision support purposes.

The Query Interface (QI) allows users to write complex queries involving temporal references between patients data, and translate these queries into FuzzyTIME low level representation. The translation process abstracts users from the low level representation and management of time. The language defined provided by QI also allows users to write sentences for the insertion and removal of temporal and atemporal data.

Two modules complete the Temporal Data Management module. Firstly, a Graphical User Interface (GUI) is necessary in order to allow non computer experts (physicians) to interact with ACUDES. Secondly, the ontology server allows QI to check the correctness of the atemporal components of inserted or removed data, i. e., only concepts and their corresponding attribute-value pair defined in domain ontology should be inserted into PTDB.

The generation of patient evolution explanations is carried out by means of three modules: Temporal Behaviour Model (TBM), Causal and Temporal Knowledge Acquisition Tool (CTKAT) and Diagnosis Agent (DA).

TBM plays the role of the knowledge base used by DA. TBM is structured as a causal network in which each disease is connected with the abnormal manifestations and other diseases caused. However, causal knowledge is not the only dimension considered in TBM. Each disease description includes knowledge about the temporal evolution of the effects related in a causal way.

CTKAT allows physicians not only to build the TBM, but to browse and manage it. In order to describe a disease, physicians can only interact with ACUDES in terms of those concepts defined in ICU ontology. This is a way of assuring the semantic consistency. The functionality provided by FuzzyTIME guarantees the temporal consistency of temporal knowledge provided by physicians.

The main function of DA is to build an explanation of the patient evolution stored in PTDB and to update it with the diagnosis conclusions reached. The explanation generated by DA must entail abnormal observations and be consistent with normal ones and with the temporal information. This interpretation of the meaning of diagnostic explanation requires a two step diagnosis task. In the first step, a temporal abductive algorithm is applied. This step builds an abductive explanation of abnormal behaviour which is temporally consistent with the information provided. The second step prunes the abductive explanation to guarantee its consistency with normal behaviour. A more detailed description of this algorithm can be seen in [5].

3 Temporal Data Management

The management of the temporal data is performed by a general purpose temporal management module call FuzzyTIME[3]. FuzzyTIME is based on a three-layered architecture which allows us to separate the interface for querying and updating temporal information (interface layer) from the layer where temporal entities and relations are managed (temporal world layer), and from their low level representation (FTCN layer). An expressive language ([3] is an extension of the one presented in [6]) which allows the formulation of complex queries involving disjunctions of relations is provided in the upper layer.

The second layer is called temporal world and contains a high level representation of temporal entities and relations. In FuzzyTIME two kind of temporal entities have been considered: time intervals and instants. Time entities can be related to each other by means of both qualitative and quantitative relations. Qualitative point-to-point, qualitative point-to-interval (both of them formalised by Van Beek [7]), Allen qualitative interval-to-interval [8] and quantitative point-to-point [9].

The third layer, which contains the low level representation of the temporal entities and relations, is based on the FTCN (Fuzzy Temporal Constraint Network) [9] formalism. A minimal network that represents the minimal constraints for temporal variables is calculated here. In this way, an efficient query answering is achieved by means of two approaches: (1) local propagation as in LATER [10], and (2) the maintenance of a minimal network. The use of fuzzy numbers as constraints allows us to make use of the Possibility Theory to solve queries about necessity and possibility, by means of a fuzzy extension of classic modal operators MAY (N) and MUST (I) which allows us to obtain a real value between zero and one as a result of a query.

3.1 Patient Temporal Database Structure (PTDB)

In the domain of application (ICU), data may be temporal or atemporal, depending on whether they are associated to time entities or not. Atemporal data are used to represent information about the patient's history, without any specific relation to time, for example, age or sex of the patient.

Temporal data include a time specification which can be specified by an absolute date or a temporal relation (by means of expression allowed in temporal language). As regards its temporal nature, information to be stored must belong to one of the following three main categories: observations -specific measurements taken on any observable patient feature-, states -produced by the temporal abstraction process over the set of observations and which represent a time interval in which the qualitative value of a feature does not change-, and events -representing the beginning and the end of a state-. A conventional approach [11] is followed for clinical history data abstraction and the technique described in [12] is applied for signal abstraction.

The different tables in the database have been designed according to the previous categories of temporal concepts. Thus, three tables are considered: observations, states and events. The basic structure of tuples is comprised of three components: concept, attribute, and value. These elements correspond to the concept being dealt with, the name of one attribute belonging to the concept, and the value of that attribute. Observation tuples include an absolute date, indicating the time at which the observation was taken, whereas a temporal reference is associated to the tuples in the state and events tables.

A graphical representation of the interrelation between the database and the internal representation of relations, points and intervals is pictured in Figure 2. Layer temporal world is represented in the middle of that figure; in that layer a high level representation of entities (intervals like *S* and points like *E1* or *E2* in the figure) and relations is maintained. This temporal world has a low level representation in the FTCN layer (see the previous section). All the data relative to observations, states and events are stored in the database and can be retrieved with a reference that has a counterpart in the entities of the temporal world.

Data Updates. There are two kinds of update operations: temporal and atemporal. First of all, and as stated in the introduction, the updates can be accomplished after the semantic consistency is checked against the domain ontology for both kinds of update. In the case of atemporal updates, the elements can be introduced into the database with a simple SQL sentence.

In the case of temporal updates a second check must be performed. Once the semantic consistency has been checked, temporal consistency of new data to be inserted with the data already stored is checked automatically by FuzzyTIME. Depending on the kind of temporal information, two different cases can be found: updating an absolute date or updating information on temporal variables. In the former case, the temporal consistency checking is not necessary, but in the latter, consistency with the previously stored in the data base must be assured.

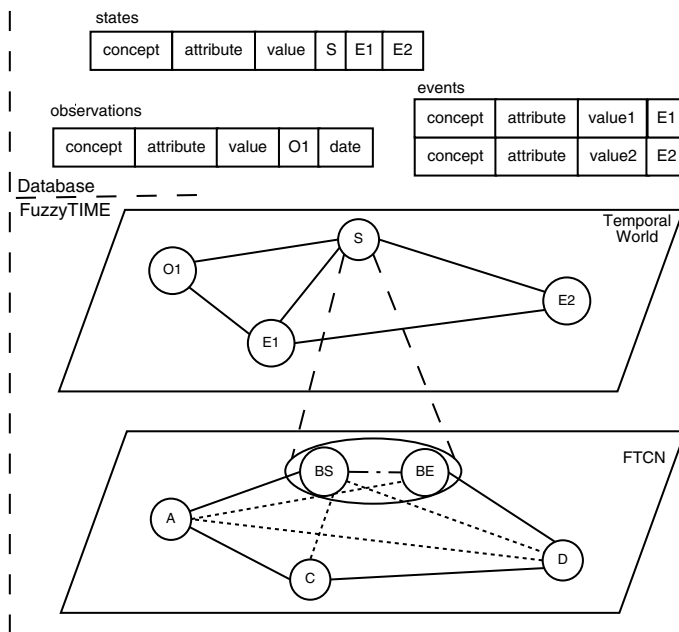


Fig. 2. Integration of FuzzyTIME with the database

Concept History Functions. Once the structure of the database has been designed, original language provided in FuzzyTIME must be extended in order to access the data stored in the database. At first sight, the definition of basic operations -such as **LAST**, **FIRST**, **NEXT**, **PREVIOUS**, and **NTH**- is needed to browse data in a single concept history. Note that a topological order is established on data belonging to the same history, but this is not the case for data corresponding to different concept histories. The argument for the **FIRST** and **LAST** functions is a tuple. That tuple can have several forms, e.g, wildcards: the tuple (concept, attribute, *) returns the first/last event (observation, or state) of the history associated with attribute; or a list of values: the tuple (concept, attribute, NOT { v_1 , ..., v_n }) returns the first/last event (observation, or state) of the history whose values do not match with any of the specified ones.

With these functions, values matching expressions such as **LAST OBSERVATION** (pain, intensity, NOT {high}) will be retrieved. In this case, the last observation of pain whose intensity is not high, that is, either moderate or low.

The remaining functions, **NEXT**, **PREVIOUS** and **NTH**, allow the user to go through a history of events, states or observations. The function **NTH** (concept, attribute, value) returns the event, state, or observation in the n th position at the history list. The functions **NEXT** and **PREVIOUS** receive a tuple (concept, attribute, value, reference) as argument and return the following/previous event (observation, or state) of the tuple in the history. **NEXT** and **PREVIOUS** can also be applied to the result of **FIRST** and **LAST** functions.

Temporal Queries. In the first instance, the temporal reasoner only accepts temporal queries [3], so we have extended the language to cope with the operations imposed by its integration in the database. In addition to the modal operators, i.e. necessity and possibility, universal and existential quantifiers have been introduced. These quantifiers allow us to deal with multiple appearances produced as result of a query to the database. Queries, in their most basic form, are comprised of two operands and a temporal relation, -one of those defined in FuzzyTIME-, and can be classified according to the type of the operands.

In the first type of queries, called level 0 queries, only the temporal entities (points and intervals) that have already been defined in the module for temporal reasoning can be included. The schema of this kind of query is:

- [MAY — MUST] (concept, attribute, value, reference) constraint [entity — date — (concept, attribute, value, reference)])

Thus, the first operand is any of the mentioned tuples (observations, states or events) plus a temporal reference; whereas the second operand can be either a tuple, with the same structure as the first operand, or the identifier of an entity or an absolute date. For example, the next query MAY (LAST OBSERVATION (pain, intensity, NOT IN {moderate})) APPROX BEFORE FIRST (fever, intensity, IN {high, moderate})) allows us to determine the degree of possibility of the last occurrence of pain whose intensity is not moderate and which happens before the first occurrence of a fever episode whose intensity is either high or moderate.

In the second type of queries, called level 1 queries, the first operand may be any tuple (observations, states or events) without a temporal relation, and any of the temporal entities, absolute date, or any tuple with a temporal reference can be used as the second operand. In this case, since the query is extended over a subset of the temporal elements that comprised the history of the event, observation, or state on which the query is performed, the universal and existential quantifiers can be used in conjunction with the modal operators MAY and MUST. This kind of queries can be formalised as follow:

- [MAY — MUST]([FORALL — EXISTS] (concept, attribute, value) constraint [entity — date — (concept, attribute, value, reference)])

As in the previous case, elements associated with a temporal reference (states, events, or observations) can be replaced by any of the functions described in section 3.1 that return an element with a temporal reference. The resolution of these queries involves an intermediate step in the translation of the query into the FuzzyTIME language since special attention must be given to quantifiers. For example, the following query MUST (EXISTS OBSERVATION(pain, intensity, high) MORE.THAN 3 DAYS BEFORE ADMISSION)) returns the necessity degree associated to the presence of an intense pain more than 3 days before the patient admission in ICU.

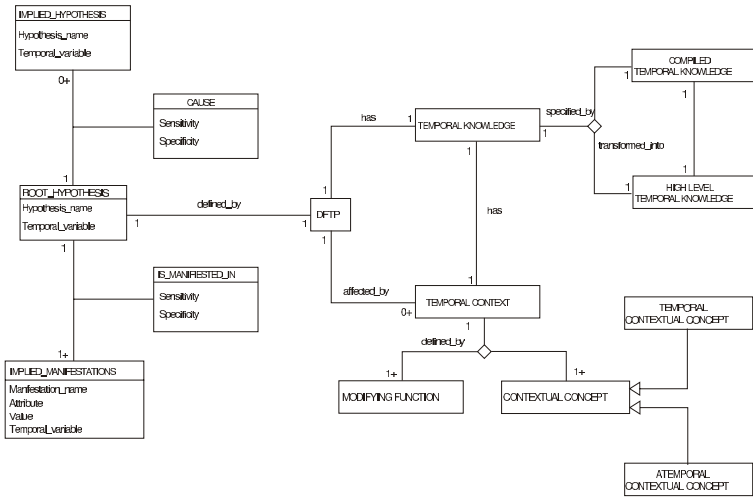


Fig. 3. Temporal Behavioural Model Ontology

4 The Knowledge Acquisition Tool

In the domain under consideration, that is the ICU, the inclusion of the temporal dimension increases the problems associated to the so called Knowledge Acquisition Bottleneck. To overcome these difficulties we have proposed:

- a temporal and causal model of diseases adapted to the requirements imposed by the domain, in which temporal relations are defined by temporal constraints between pattern elements [2,13].
- an architecture for a knowledge acquisition tool that guides the model building process thus reducing expert cognitive load in defining the model.

4.1 Temporal Behavioural Model

The causal and temporal model is adapted to the requirements imposed by the diagnosis agents designed for decision support purposes in the ICU domain. However, we have approached the TBM design process from a domain independent point of view, so as to provide useful knowledge components for potential knowledge reuse in other domains. This work is focused on the underlying knowledge structured for the diagnosis task, which is analysed in greater depth in [13], and which addresses the problems associated to the so called knowledge acquisition bottleneck.

Figure 3 depicts the ontology defining the underlying structure of the KB. As can be deduced from this figure, the key element in the TBM is the *Diagnostic Fuzzy Temporal Pattern* (DFTP hereinafter). A DFTP captures the causal and temporal relations between an hypothesis and its effects.

A DFTP is associated to a **ROOT HYPOTHESIS**, which is defined through the hypothesis name and a temporal variable (for example, (Retrograde_Cardiac

`_Insufficiency,t1)`), which will be used in the temporal specification of the DFTP. A root hypothesis is associated to at least one **IMPLIED MANIFESTATION** and zero or more **IMPLIED HYPOTHESES**, which allows us to capture the fact that there are diseases that can cause other implied diseases. Implied hypotheses are specified in the same way as root hypotheses, whereas implied manifestations are defined by the manifestation name, and an attribute-value pair (for example `(pain, location, precordial,t2)`). Finally, both causal relations **IS.MANIFESTED.IN** and **CAUSE** require two attributes in order to capture the specificity and sensibility measures used in Evidence Based Medicine.

Apart from the causal knowledge, a DFTP requires temporal and contextual knowledge. **Temporal knowledge**, in both DFTP definition and context specification has to be described at two different levels of detail:

- Firstly, we need to keep the high level specifications (**HIGH LEVEL TEMPORAL KNOWLEDGE**) of temporal constraints, as provided by physicians in a high language design for this purpose (`(pain, location, precordial,t1) APPROX 5 MIN AFTER Retrograde.Cardiac.Insufficiency,t2)`).
- Secondly, this high level representation must be translated into the FTCN (Fuzzy Temporal Constraint Networks) formalism [6], conforming the **COMPILED TEMPORAL KNOWLEDGE**. The FTCN formalism allows us to apply temporal reasoning processes which:
 - can infer the complete set of temporal constraints between pattern elements, without the need to oblige the physician to define all the temporal constraints.
 - can detect inconsistencies in the temporal information provided by physicians.

Finally, a DFTP can be affected by zero or more **TEMPORAL CONTEXTS** (a disease effect could be different if, for example, some drugs are prescribed to the patient, or some risk factors are present). Temporal contexts are specified by at least one **contextual concept**, which may have associated temporal knowledge (i.e., those contextual concepts describing drugs prescriptions) or may not (for example, those **contextual concepts** describing risks factors associated to patients). A formal specification of TMB can be found in [2,13].

4.2 ICU Domain Ontology

The ontology for the ICU domain was built using Protégé-2000 [14] and the Unified Medical Language (UMLS) [15] as standard reference of vocabulary terms. Other ontologies have been considered for reuse such as EON ontology [16].

ICU-ontology represents the taxonomy regarding those concepts used in disease descriptions. Relations other than the taxonomic are not necessary for our purposes. Causal relations between ICU-ontology can be achieved by the corresponding mapping with the TBM ontology concepts. The mapping wizard provided by the KA tool facilitates this process.

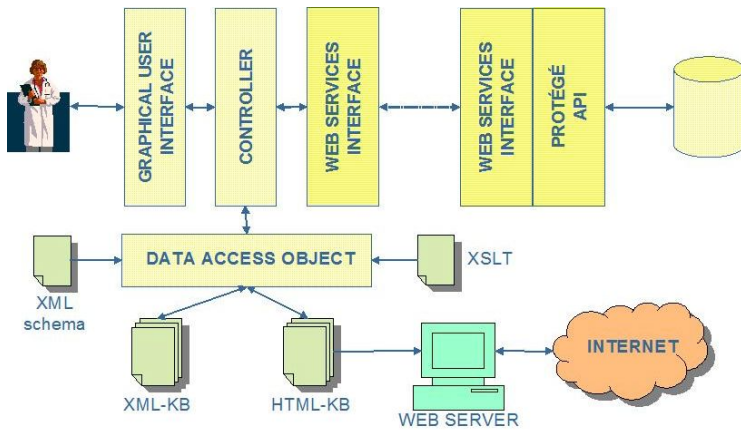


Fig. 4. Knowledge Acquisition Tool Architecture

4.3 The KA Tool Architecture

The architecture for the KA tool is shown in Figure 4. The main module of this architecture is the Dialogue Controller. The Dialogue Controller is composed of a set of wizards which defines sequences of steps to be accomplished for the acquisition of the different knowledge categories, as specified in the TMB ontology, through the Graphical User Interface. An important part of the controller is the module in charge of time representation and management, which imports the required functionalities from the FuzzyTIME temporal reasoner.

In the current version of the tool the KB has been implemented in XML. Nevertheless, the Data Access Object pattern (DAO) constitutes a data abstraction layer that isolates the tool from the physical representation of the KB. If other representations languages are considered in future versions, this is the only module that will have to be modified.

The XML version of the Data Access Object requires an XML schema to create the different XML files defining the temporal patterns. The XML schema represents the internal structure of DFTP and is created from the TBM ontology. Once the XML files are created, they can be automatically translated to HTML files via the corresponding XSLT files and published on the www as soon as patterns are defined.

Finally, the controller is connected via the corresponding web services interface to the ontology server. The ontology server is based on the API provided by the Protégé-2000 ontology edition tool [14], and makes the mapping between TBM ontology concepts and domain ontology concepts possible, via the mapping wizard. This allows the controller to present only those domain concepts that might be chosen by the physician in each step of the KA process, thus reducing the cognitive load on the expert side. Of course, new concepts can be added to the domain ontology. The controller is in charge of inserting new concepts in the right place in the domain ontology. However, ontology structural changes need to

be accomplished from outside the tools. KA for other domains can be accomplished by this tool, whenever the new domain meets the requirements imposed by the TBM ontology. A new domain will require to load the corresponding ontology in the ontology server and to define new mappings between the ontology and the TMB ontology.

5 Conclusions

In this work we have presented ACUDES, an decision support system for ICUs. In order to cope with the importance of the temporal component in the information regarding the patient evolutions, ACUDES provides techniques for managing and representing time.

An important contribution is the use of ontologies to support all ACUDES functionalities. Firstly, the ICU ontology gives an underlying structure to the knowledge-base, which simplifies the knowledge acquisition process, and taking into account that is based on UMLS terminology, ICU ontology becomes a standard framework in which knowledge can be shared. Secondly, ICU ontology supports the semantic consistency of the PTBD, allowing only interactions in the terms defined in the ontology.

As pointed out above, due to the huge amount of data involved in this system, data are stored in a data base. In this paper we have dealt with the integration of a general purpose module for temporal reasoning, FuzzyTIME, with a database where the domain information is stored. The three-layered architecture of FuzzyTIME and the general structure of the data base allow a seamless integration with any other domain.

The architecture can benefit from (1) the major features of the temporal reasoner, such as the ability to deal with qualitative and quantitative temporal constraints and the efficient query answering process, and (2) the ability of the data base manager for managing large amounts of data. The result obtained in this way is a system able to perform operations on temporal qualitative or quantitative constraints, such as asserting new temporal constraints, checking the consistency of constraints, and inferring new temporal constraints.

As regards the query language and the interaction with database. Section 3.1 provides basic functions for browsing concept histories and for retrieving a specific occurrence. Furthermore, the kind of queries formerly allowed by FuzzyTIME have also been extended to take advantage of these new functions, making it possible to include existential and universal quantifiers in queries involving both temporal and atemporal information.

This work also addresses the problem of knowledge acquisition in domains where causal knowledge is specified along the temporal dimension. This causal and temporal knowledge can be captured by the TBM in which the temporal evolution of diseases is represented. The KA-tool provides a mapping wizard that facilitates the specification of the ontology mapping between the TBM ontology and the domain ontology. Other wizards define the sequences of steps that have to be accomplished for the acquisition of the different knowledge categories, as

specified in the TMB ontology, through the Graphical User interface. Ontology mappings ensure that only those domain ontology concepts that might be selected in each KA step are presented to the expert. This functionality reduces the user cognitive load in building the KB.

References

1. Horn, W.: AI in Medicine on its ways from Knowledge-Intensive Systems to Data-Intensive Systems. In: *Artificial Intelligence in Medicine*. Volume 23. (2001) 5–12
2. Palma, J., Marín, R., Campos, M., Cárceles, A.: ACUDES: Architecture for Intensive Care Units DEcision Support. In: *Conference Proceedings of the second joint EMBS-BMES conference*. ISBN: 0-7803-7613-. (2002) 1938–1939
3. Campos, M., Cárceles, A., Palma, J., Marín, R.: A general purpose fuzzy temporal information management. In: *EurAsia-ICT 2002. Advances in information and communication technology*. ISBN: 3-85403-161-3., Teherán, Irán (2002) 93–97
4. Koubarakis, M., Skiadopoulos, S.: Querying temporal constraint networks in PTIME. In: *Proceedings of the 6th National Conference on Artificial Intelligence (AAAI-99)*, Menlo Park, Cal., AAAI/MIT Press (1999) 745–750
5. Palma, J., Marín, R., Sánchez, J., Palacios, F.: A model-based temporal abductive diagnosis model for an intensive coronary care unit. In S. Barro, R.M.e., ed.: *In Fuzzy Logic in Medicine. Studies in Fuzziness and Soft Computing*. Volume 83. (2002) 205–235
6. Barro, S., Marín, R., Mira, R., Patón, J.: A model and a language for the fuzzy representation and handling of time. *Fuzzy Sets and Systems* **61** (1994) 153–175
7. van Beek, P., Cohen, R.: Exact and approximate reasoning about temporal relations. *Computational Intelligence* **6** (1990) 132–144
8. Allen, J.F.: Maintaining knowledge about temporal intervals. In Brachman, R.J., Levesque, H.J., eds.: *Readings in Knowledge Representation*. Kaufmann, Los Altos, CA (1985) 509–521
9. Marín, R., Barro, S., Palacios, F., Ruiz, R., Martín, F.: An approach to fuzzy temporal reasoning in medicine. *Mathware & soft Computing* **3** (1994) 265–276
10. Brusoni, V., Console, L., Terenziani, P.: Efficient query answering in LaTeR. In: *TIME-95 International Workshop on Temporal Representation and Reasoning*. (1995) 121–128
11. Shahar, Y.: Efficient algorithms for qualitative reasoning about time. *Artificial Intelligence* **90** (1997) 79–133
12. Felix, P., Barro, S., Marín, R.: Fuzzy constraint networks for signal pattern recognition. *Artificial Intelligence. Special Issue: Fuzzy set and possibility theory-based methods in artificial intelligence* (2003 (In Press).)
13. J.Palma, R.Marín: Modelling contextual meta-knowledge in model based diagnosis. In: *Proceedings of the ECAI-2002*. (2002) 407–411
14. Genneari, J., Musen, M.A., Ferguson, R.W., Grosso, W.E., Crubezy, M., Eriksson, H., Noy, N., Tu, S.W.: The evolution of protégé: An environment for knowledge-based systems development. *International Journal of Human-Computer Interaction* (2002) In press
15. National Library of Medicine: Unified Medical Language System 13 Edition. (2002)
16. Tu, S.W., Musen, M.A.: Modelling data and knowledge in the eon guideline architecture. Technical Report SMI-2001-0868, Stanford Medical Informatics (2001) <http://www-smi.stanford.edu/pubs/SMI.Reports/SMI-2001-0868.pdf>.

Development of a Scalable, Fault Tolerant, and Low Cost Cluster-Based e-Payment System with a Distributed Functional Kernel*

C. Abalde, V. Gulías, J. Freire, J. Sánchez, and J. García-Tizón

LFCIA Lab, Campus de Elviña
15071, La Coruña, Spain

{carlos,gulias,freire,juanjo,tizon}@lfcia.org
<http://www.lfcia.org>

Abstract. In this paper is presented the design and implementation of a payment gateway. It acts as a mediator among an e-Commerce, a final customer computer, mobile phone or other network access device and a bank host. The gateway was designed using modern software engineering tools (such as unified modeling language, design patterns, etc.), implemented using a functional language (ERLANG/OTP) and supports the typical features of other common payment gateways. The design and implementation pays a lot of attention to its architecture (using distributed computing to improve service availability -scalability, fault-tolerance-) and to the chance of apply formal verification tools and techniques to improve system design and performance, and to ensure system correctness. The architecture and ideas proposed in this paper are intended to be a first step towards the implementation of other and more complex payment systems.

1 Introduction

In this paper are described the design and implementation of an electronic payment system over an open network such as Internet, based on an innovative architecture that can be reused for other similar systems and applications. The proposed system can be divided into three main components:

1. The *payment gateway system*, which is the main focus of this work, is a distributed application implemented using a functional language (ERLANG/OTP) and deployed in a Beowulf cluster of computers.
2. The *point of sale (POS) system*, an interaction facade library for the payment gateway system. It has been implemented in several languages (PERL, C++ DLL and JAVA) in order to make easier the integration with the existent e-commerce products.
3. The *host adapter*, used by the payment gateway for the access to the financial networks. The adapter acts as an abstract layer that hides the concrete details of each financial entity host.

* Partially supported by MCyT Project TIC 2002-02859.

The payment gateway is the main and more complex system. Its key features are:

- *Execution in a cluster of computers*, using a schema describing how the system load is going to be distributed among the cluster nodes.
- *Scalable*. A new set of nodes can be added to the cluster without changing the system implementation and without stopping it. The system is able to satisfy the requirements in a simple context (downwards scalability); and is able to increase the performance by adding new resources to the architecture, serving more concurrent users (upwards scalability).
Unscalability is a common problem in similar systems, where the payment gateway is normally a performance bottleneck; some of these systems are not scalable at all, and others design complex mechanisms for trying to scale.
- *Fault tolerant*, meaning that if one of the nodes of the system architecture crashes, the users are not going to notice any problem with the service. Whenever this happens, another node is going to assume the role of the crashed in a transparent way. With 24x7 expected uptime, there is a need for some kind of mechanisms to keep working at least in a degraded mode when some kind of fail happens.
- All the internal state of the system is stored into a *distributed database* whose configuration and distribution ensures that the distribution of the database is not going to be a limitation for the gateway scalability.
- The system *is able to interact with different kind of devices (multi-device access)*, such as Web browsers, WAP terminals, or any other that could appear in the future, without changing the internal implementation.
- *Personalization* of the contents that are shown to the clients depending on the needs of each virtual shop contracting the services of the payment gateway.
- *Low cost*. The goal is to satisfy all the above mentioned requirements with an architecture that can reduce the implementation and deployment costs.

The paper is structured as follows. In section 2, the state of the art and motivations for the proposed system are described. In section 3, the key technologies used during the design and development of the system are discussed. Then, a first approach to the architecture of the payment gateway is described in section 4. Section 5 explains more in detail some of the refinements and improvements of the system. Finally, section 6 shows some performance measurement results.

2 State of the Art

Traditional means of payment suffer from various well-known security problems: money can be counterfeited, signatures can be forged and checks can bounce. Electronic means of payment (electronic payment systems) retain the same drawbacks. In addition, they entail some additional risks: unlike paper, digital “documents” can be copied perfectly and arbitrarily often; digital signatures can be

produced by anybody who knows the secret cryptographic signing key; a customer's name can be associated with every payment, eliminating the anonymity of cash, etc [1]. Nevertheless, a well designed electronic payment system can be as secure as the traditional ones, or even more, as well as being more flexible.

Since mid-nineties, dozens of electronic payment systems which use public insecure networks as Internet had been proposed. Only some of them have been implemented. A classification of the more relevant systems and projects could be done using the following categories [1,2,3,4]:

- *Support for the use of card numbers in public networks: ad hoc* solutions based on SSL, Net Market, CyberCash, SET...
- *Electronic intermediates*: VirtualPIN, NetBill, VirtualCash, PayBox, Yahoo! PayDirect, PayPal, MobiPay...
- *Electronic checks*: FSTC Electronic Check Project, NetCheque...
- *Electronic cash*: CAFE, Mondex, NetCash, CyberCoin...
- *Micropayments*: Millicent, SubScrip, MicroMint...

The payment gateway proposed in this paper falls in the first category, but with an innovative kernel architecture comparing with other solutions with similar functionality. This architecture is intended to be a first step towards the implementation of other payment systems.

3 Key Technologies

Besides the innovative architecture presented, three of the most important and differentiating features of the proposed solution are the use of a distributed functional language for the implementation, the adaptation to a LINUX cluster-based architecture and the intensive use of design patterns.

3.1 Erlang/OTP

The programming language used for the implementation of the payment gateway, ERLANG/OTP [5,6], has been designed and used in Ericsson's computer science lab (CSLab) for developing distributed control systems. The combination of the functional paradigm and the parallel computation defines a declarative language, with almost no side effects and with a high level of expressiveness, abstraction and ease for prototyping.

ERLANG/OTP is specially suited for soft real time, distributed and fault tolerant systems. The language is based on asynchronous message passing, transparent communication of values, high order communications, and it is designed to support a high amount of concurrent processes. The language is suited for distributed systems development, and allows transparent allocation of processes among different nodes. Besides, ERLANG/OTP includes primitive functions for supporting fault tolerance and allows code replacement without having to stop the system.

In addition to the language itself, the proposed solution uses intensively the libraries and distributed design patterns of the *Open Telecom Platform* (OTP), which includes generic servers, supervision mechanisms, a distributed database (Mnesia) with transparent location, fragmentation and replication, fully integrated with the language, and several utilities such as a C interface or an HTTP server with support for SSL [7] secure connections (INETs).

All these characteristics become ERLANG/OTP a very well suited language to the payment gateway development.

3.2 Linux Cluster

The use of Beowulf clusters (low cost LINUX-based architecture) as part of the proposed solution is one of the keys that makes the approach innovative.

The main advantages of LINUX technology are the existence of an Open-Source community with experience in high speed networks, distributed systems and clustering, the availability of the source code and development tools, the ease for migrating to other UNIX versions, and the support for several platforms.

These characteristics, added to the low cost and flexibility of Beowulf clusters, are keys to reach the goals proposed in section 1.

3.3 Design Patterns

The design patterns are a concise and elegant way for representing architectural concepts that are commonly used by experienced software designers. They give the designers a common reference framework and a common terminology, being useful for improving the understanding and for reusing the knowledge acquired in past successful developments.

The concept has been borrowed from the civil architecture [8] and has been a revolution in the way of understanding the software design, specially after the collection published by a group of authors known as the *gang of four* (GoF) [9].

Typical ERLANG/OTP applications are designed as a big number of small concurrent processes. Despite of the simplicity of individual processes, the global behavior of the ERLANG/OTP system can be very complex. This idea suggests a decomposition guided by the use of design patterns, from the adaptation to a distributed platform of the classical GoF patterns to other low-level language-specific patterns, available in this case as *behaviors* (generic servers, supervision tree, generic finite state machine, etc.).

In the system design, UML [10] was used as modelling language for defining the interactions among the agents (sequence diagrams) and the changes in the system state due to the transactions (state diagrams). Design patterns have been used at modelling time for dealing with the traditional problems of distributed architectures.

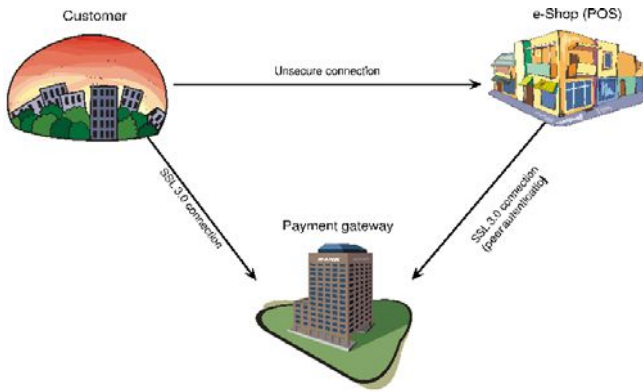


Fig. 1. Agents of the system

4 Design and Architecture Overview

The payment gateway went through several refinement steps, from the initial prototype, satisfying most of the requirements identified during the analysis, to a final system with the desired features of distribution, scalability and fault tolerance, all with a reasonable low cost architecture.

4.1 Interaction among the Agents

Starting with the basic interaction among the three main agents in the payment system (POS, payment gateway and customer - figure 1), going through a phase of requirements capture, a sequence, XML [11] structure and content of the exchanged messages using UML sequence diagrams. Afterwards, state diagrams were used for formalizing the state changes of an electronic transaction, both in the POS and the payment gateway.

The outcome was a communication protocol among the agents, where different cryptographic techniques (symmetric and asymmetric cryptography, digital envelopes, etc.) and XML messages digitally signed, encapsulated and transported using HTTP secure connections (SSL) were used.

XML was chosen for representation of the messages to simplify the interoperability of the agents, and because it is a newer, more flexible and clearer alternative to ASN.1, used in systems like SET [12]. All the messages follow the structure depicted in figure 2, and include the digital signature of the sender, calculated using a simplification of the DOMHASH algorithm [13].

4.2 Separation between Content and View

With the goals of making the gateway able to integrate the design of the contents showed to the customers with the design of the virtual shop; and allowing the use of different access devices (WEB, WAP, etc.), the gateway was designed for

```
<message id="..." version="1" revision="0" >
  <header>
    <trID> (...) </trID>
    <ReqResID> (...) </ReqResID>
    <nonce> (...) </nonce>
  </header>
  <body> (...) </body>
  <signature> (...) </signature>
</message>
```

Fig. 2. Basic message structure

```
<digitalEnvelope>
  <key algID="dsa" >
    (...)
  </key>
  <content algID="bf/cbc" >
    (...)
  </content>
</digitalEnvelope>
```

Fig. 3. Digital envelope structure

generating intermediate results as XML content, that is translated to a final format (HTML, WML, etc.) using XSL style-sheets [14], selected using a rule-based system.

4.3 Host Adapter

The access of the payment gateway to the financial network is done by means of we have called “the host”. With the goal of abstracting away from the concrete details of the host used by each financial entity, the access interface is proposed as a strategy pattern [9], implemented as an interchangeable module that is part of a generic server state. This module acts as an access and translation point to the host.

4.4 Initial Prototype

Putting away the POS, because it is a simpler software component that acts as facade for the payment gateway, the combination of all the ideas introduced in this section gave place to the first prototype of the gateway. In this initial version, still not distributed and with reduced fault tolerance features, the same technologies that in the final version (ERLANG/OTP, MNESIA distributed database, XMERL XML parser, the XSL processor SABLON, the HTTP server INETS and the cryptographic library OPENSSL) were already used, and all the requirements identified during the analysis phase were already covered.

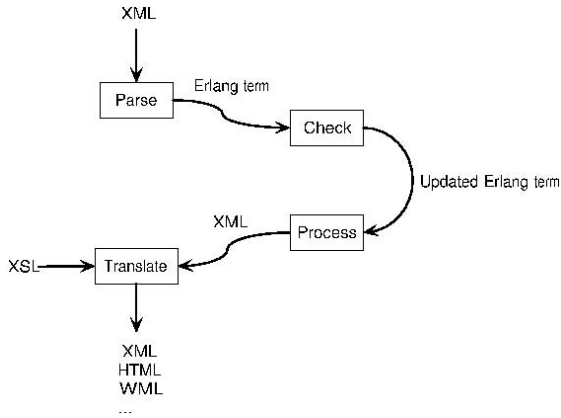


Fig. 4. Steps for processing one request

5 Design Refinement

The next step was to evolve the initial prototype of the payment gateway to a flexible, scalable and fault tolerant system.

5.1 Decomposition into Subsystems

The initial gateway, with a monolithic architecture, evolved towards a new version decomposed into three large subsystems:

- *Virtual shops access subsystem*, composed by a pool of nodes, each of them executing a lightweight HTTP server and with specific access controls (access lists, SSL authentication of the clients, etc.) for replying POS's queries.
- *Customers access subsystem*, composed by a pool of nodes, each of them executing a HTTP server which answers the customers requests. Both this subsystem and the previous one can be composed by more than one node. The load balancing schema is out of the scope of the payment gateway, being a simple reasonable option to use a round robin DNS giving a unique point of access from the outside.
- *Transaction processing subsystem*, composed by a pool of nodes in charge of processing the transactions. This subsystem has been decomposed into a variable number of subsystems, in order to avoid that the use of the distributed database could cause scalability restrictions in the gateway (section 5.3). Each of these subsystems is composed by a variable number of nodes, balancing the load among them following a master-slave architecture, where each slave decomposes the processing of each request into the steps shown in figure 4.

From a practical point of view, each node of the cluster can carry out the role of one or more subsystems. Besides, each subsystem is composed by a vari-

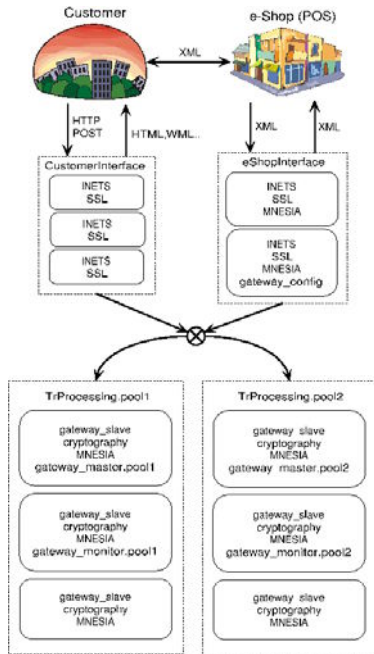


Fig. 5. Example configuration of the payment gateway

able number of nodes, that can be modified changing the configuration files; or without stopping the system using the web administration tools of the gateway.

5.2 Fault Tolerance

The final version of the payment gateway covered the initial goal of the fault tolerance, with features such as the database distribution, the subsystems with redundant nodes, or the distribution of the critical applications.

It is important to make clear that, besides the processes of each subsystem, other distributed applications carrying functions as monitorization or load balancing are being executed in the cluster at the same time. This critical applications had been configured following a process supervision tree, and they are monitorized by the ERLANG/OTP kernel in order to guarantee the high availability of the system, even in the case of loosing some of the cluster nodes.

5.3 Scalability

The final prototype can scale in several dimensions, depending on the deployment needs.

First of all, the virtual shop access and customer access subsystems have unlimited scalability.

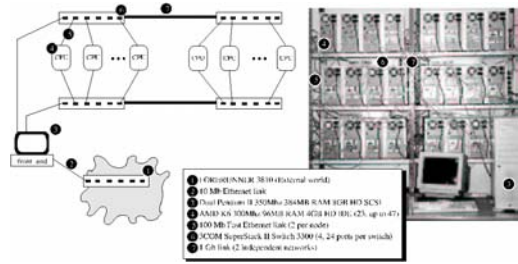


Fig. 6. LFCIA's cluster (*Borg*)

Secondly, new transaction processing subsystems can be added with no limitation to the system.

And finally, Each transaction processing subsystem can scale independently, with the unique restriction of the restrictions related to the distributed database. The writing actions in a distributed database with a high number of nodes can be too slow, due to the need of locking all the other nodes with writing access to the involved tables. The transaction processing subsystem writes very often to the database in order to maintain the transaction state updated, and this could produce performance limitations. In that case, instead of adding new nodes to the existent subsystems, it would be a better idea to add a new processing subsystem with the new nodes, thus doing a partition in the distributed database.

Figure 5 illustrates one of the possible configurations of the payment gateway. It shows one virtual shop access subsystem with two nodes, one customer access subsystem with three nodes, and two subsystems for transaction processing, each of them composed by three nodes. This configuration could be deployed in a cluster with between 3 and 11 physical nodes.

6 Performance

In order to check some of the features of the system, some benchmarks have been done in LFCIA's cluster, *Borg* (figure 6). This cluster is composed by 23 internal nodes and a front end, all of them connected by a double Fast Ethernet 100 Mbps network (channel bonding) with four 24 port switches. The switches themselves are connected by a 1 Gbps network.

Leaving aside the tests of availability, fault tolerance and scalability, we will focus in the performance tests. For the accomplishment of the experiments some scripts, launched at different nodes, were used to simulate different workload settings. In addition the SSL connections were replaced by unsecure connections and the access subsystems were configured with enough nodes (distributing the load using a round robin DNS) to avoid a bottleneck in the access to the transactions processing subsystem, which is the focus of the performance tests. From this basic scenario the transaction processing subsystem was tested with different deployments over the cluster.

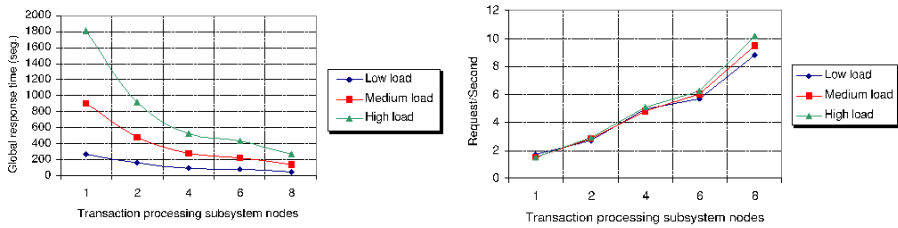


Fig. 7. Performance with different load scenarios

Figure 7 shows how the incorporation of new nodes to the processing subsystem originates a linear reduction of the response time and, therefore, an increase of the number of requests processed by second. Additionally, the figure shows that there is a minimal processing time by message that cannot be reduced adding nodes to the subsystem. This time only can be reduced optimizing the processing steps (parsing, translation, optimizing cryptographic operations, etc.).

7 Conclusions and Future Work

The payment gateway implementation covered the expectations of its design. The use of ERLANG/OTP as the implementation language was a good decision to build a scalable and distributed system, with an innovative architecture that can be reused to develop more complex payment systems.

The ERLANG/OTP alternative could have been PVM or MPI, which allow the implementation of distributed systems, but to make them scalables it is not an easy task. Another option could be the use of a distributed object technology such as CORBA or RMI. With a good design we can build a distributed and scalable system, but its implementation would be more complex for sure.

Additionally, choosing of ERLANG/OTP as the implementation language gives to the payment gateway support to uninterrupted operation and fault tolerance, in a simple way. Moreover, another advantage of using ERLANG/OTP as the underlying platform for the system development, is that techniques coming from the formal methods area can be applied in order to improve the system design and performance [15], and to ensure (at least partially) the system correctness [16]. These techniques can be used more naturally when taking as input a high level declarative language as ERLANG, and their results are specially appreciated in security critical applications as a payment gateway.

Finally, the combination of XML and XSL stylesheets to the communication between systems has showed to be the best option to simplify its interoperability and to support tasks such as the content personalization and the independence of the access protocol.

Independently of the improvements over the specific payment system, the architecture could be extended and improved in aspects such as the load distri-

bution among the cluster nodes, the improvement of the performance in some parts of the system, the support of hardware cryptographic devices, the refinement of the personalization features or the the application of any XML digital signature standard [17].

References

1. N. Asokan, Phil Janson, Michael Steiner, and Michael Waidner. State of the art in electronic payment systems. In Marvin V. Zelkowitz, editor, *Advances in Computers*, volume 43, pages 425–449. Academic Press, March 2000.
2. Dennis Abrazhevich. Classification and characteristics of electronic payment systems. In *Proceedings of Electronic Commerce and Web Technologies*. Springer, September 2001.
3. ePayments Systems Observatory. ePSO inventory database. <http://epso.jrc.es>, 2002.
4. Donald O'Mahony, Michael Peirce, and Hitesh Tewari. *Electronic Payment Systems*. Artech House, 1997.
5. Erlang/OTP Team. Erlang/OTP documentation. <http://www.erlang.org>, 2002.
6. J. L. Armstrong, M. C. Williams, C. Wikström, and S. R. Virding. *Concurrent Programming in Erlang*. Prentice Hall, 2nd edition edition, 1996.
7. OpenSSL Team. OpenSSL documentation. <http://www.openssl.org>, 2002.
8. A. Alexander. *A Pattern Language: Towns, Buildings, Constructions*. Oxford University Press, 1977.
9. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Software*. Addison Wesley, 1995.
10. Martin Fowler. *UML Distilled*. Addison-Wesley, 1997.
11. T. Bray, J. Paoli, C. M. Sperberg-McQueen, and E. Maler. *Extensible Markup Language (XML) 1.0*. 2nd edition edition, October 2000.
12. Visa and MasterCard. *SET Secure Electronic Transaction. Book 1: Business Description*. SETCo, May 1997. Version 1.0.
13. H. Maruyama, K. Tamura, and N. Uramoto. RFC 2803: Digest values for DOM (DOMHASH), 2000.
14. James Clark. *XSL Transformations (XSLT) 1.0*. November 1999.
15. Thomas Arts and Clara Benac Earle. Verifying Erlang code: a resource locker case-study. In *Int. Symposium on Formal Methods Europe*, volume 2391 of *LNCS*, pages 183–202. Springer-Verlag, July 2002.
16. Thomas Arts and Juan José Sánchez Penas. Global scheduler properties derived from local restrictions. In *Proceedings of ACM Sigplan Erlang Workshop*. ACM, October 2002.
17. John Boyer. Canonical XML 1.0. <http://www.w3.org/TR/xml-c14n>, March 2001.
18. G. Winfield Treese and Lawrence C. Stewart. *Designing Systems for Internet Commerce*. Addison-Wesley, Reading, MA, USA, 1998.

Generative Communication with Semantic Matching in Distributed Heterogeneous Environments^{*}

Pedro Álvarez¹, José A. Bañares¹, Eloy J. Mata², Pedro R. Muro-Medrano¹,
and Julio Rubio²

¹ Department of Computer Science and Systems Engineering, University of Zaragoza,
María de Luna 3, E-50015 Zaragoza (Spain).

{alvaper,banares,prmuro}@posta.unizar.es

² Department of Mathematics and Computer Science, University of La Rioja,
Edificio Vives, Luis de Ulloa s/n, E-26004 Logroño (La Rioja, Spain).

{eloy.mata,julio.rubio}@dmc.unirioja.es

Abstract. Different standard middleware proposals have emerged to provide computing models and communication among components in open distributed systems. Nowadays, Internet is becoming an increasingly relevant alternative to middleware platforms, due to the success of Web services in solving problems of application-to-application integration in distributed and highly heterogeneous environments. However, a coordination model is necessary to build open and flexible systems from active and independent distributed components. In this paper, we present a Web-enabled Coordination Service to orchestrate heterogeneous applications based on the Generative Communication model with semantic matching. Our aim is to use Internet as a real distributed computing platform, considering heterogeneous semantic interoperability.

1 Introduction

Traditional middleware platforms (such as CORBA, COM or EJB) are sometimes presented as a general solution for distributed computing in heterogeneous contexts. Nevertheless, this is not completely true in practice. On one hand, they are based on object-oriented constructs and then some degree of homogeneity is required, at least from a programming-paradigm point of view. On the other hand, the physical substrate, on which communications are established, is abstracted. (Note that these two features of middleware are not considered negative for us; they simply imply certain consequences that are not always explicitly stated when these general platforms are introduced.) Internet is becoming an increasingly relevant alternative to standard middleware platforms, due to the success of Web-services in solving problems of application-to-application integration in distributed and highly heterogeneous environments. These Web-services

^{*} Partially supported by the Spanish Ministry of Science and Technology through projects TIC2000-1568-C03-01, TIC2000-0048-P4-02, TIC2002-01626.

may be implemented on different computational models, and communicate and interchange data among them using standard Internet protocols and data formats, such as HTTP and XML, respectively. However, there is no standard support enabling these distributed services to work together harmoniously and in a *coordinated* way (this is one of the main difficulties inherited from traditional approaches). To enable this cooperation among distributed services it is necessary to take into account two essential aspects of Internet. First, from a technical point of view, communication by means of HTTP is always synchronous (via sockets, for instance), but in distributed and concurrent computing asynchronous communication is also mandatory. The second aspect has to do with the *real* behaviour of Internet: it is a hostile medium where the reliability of the communications is poor and unsafe.

In this paper, a proposal to overcome some of these difficulties, by means of a Web-services based approach, is presented. Our aim is to use Internet as a real distributed computing platform, considering semantic interoperability among systems developed in very different programming languages, different even from a programming paradigm point of view. (It is worth noting that we do not consider our proposal as an alternative to standard middleware; it is rather an experience linked to a set of ideas largely orthogonal to any particular implementation tool.)

The basic idea is to use a *coordination model* that acts as a kind of “glue” gathering separated activities into a single computing device. Instead of starting from scratch, the model known as Generative Communication has been chosen, and more precisely the so-called *Linda* model [4]. Linda is a very abstract artifact based on two notions: tuple and tuple space. The tuples are extracted from the tuple space by means of a pattern matching process. Our proposal is based on this obvious observation: if the simple matching strategy of Linda is replaced with a *complex matching*, then very general kinds of interoperability can be achieved. To this aim, we work with a version of Linda where the tuples admit a description by means of attribute/value pairs, which is similar to a XML-data definition, extracted from [5]. Now, Linda can be used in this XML context, providing a promising way to coordinate, communicate and collaborate on the net.

These ideas, which can be summarized as “Linda with semantic and structured matching, using XML”, have been put into practice to deal with Web-services, but trying to avoid the two difficulties previously mentioned: the synchronous and hostile nature of Internet. Our implementation uses JavaSpaces as core. (JavaSpaces is a Linda realization based on the Jini technology.) This saves us the rewriting of routine code to manage tuples and tuple spaces. We have enriched JavaSpaces to work with XML-entries. Nevertheless, note that JavaSpaces only provides the “top level”, non-structured (and, of course, non-semantic) matching. From this extension of JavaSpaces, a Web Coordination Service has been developed, achieving the following objectives:

1. Coordination operations are accessible through the HTTP service interface, using a XML data format to specify the exchanged messages.

2. We have obtained a uniform way to deal with complex matching (both semantic and structured).
3. We have included an event-based asynchronous communication over HTTP, generated through PushLet technology. This communication style has allowed us to incorporate reactive coordination aspects to the Generative Communication model (by means of a system of subscriptions).
4. We have created internal agents/proxies to represent the external Web-services and cooperate on behalf of them. This reduces the number of Internet connections and hence minimizes the problems of reliability.

The paper is structured as follow. Section 2 shows a model for coordinating open Web services. Section 3 presents a description of the Web-enabled Coordination Service (WCS). The WCS consists of three software components: XML-based Space, Java Coordination Component and HTTP Coordination Component. In Section 4, the semantic pattern matching is described. The matching is made in two steps and some semantic resources, such as thesauri and ontologies, are used. Section 5 tries to formalize the operations of the coordination model and the complex matching process of our approach. In Section 6 we show two application examples: the Dining Philosophers problem, as an academic example; and a real project in the context of location-based services and automatic vehicle-monitoring (to orchestrate different OpenGIS and LIF Web services). Finally, conclusions and future work are presented.

2 A Model for Coordinating Open Web Services

Web Services are self-contained, modular applications that can be described, published, discovered and invoked over a network. Then, the Web service based approach is an application integration concept, and some essential elements should exist to support it:

- Component interfaces must be *Web-enabled*, providing a collection of functions that are packaged as a single entity and published to the network to be used by other programs. These functions must be specified according to wide-accepted standards that ensure interoperability, ease of use, and loose coupling of Web services.
- It is necessary a universal *middleware for interchanging data* that ensure interoperability beyond specific distributed computing platforms (e.g., COM, CORBA, EJB) and/or different programming languages. Web services communicate using HTTP and XML. Therefore any device, which supports these technologies, can both provide and access Web services.
- A *coordination model* that acts as the glue that binds separate activities into an ensemble. Distributed applications and Web services are a natural breeding ground for heterogeneity. A service that needs contributions from different machines, providers and computing models requires a natural means for multi-language. Therefore, it is necessary to provide a coordination model to represent these interactions and the space where they happen.

In this context, each Web service can be seen as a type of individual entity, which needs to communicate and synchronize with another Web services, or entities distributed over Internet to get together a response for the user's request. It is necessary to emphasize that Internet is a highly heterogeneous cooperation environment. Many Web services are developed in different programming languages, are running over different executing environments or/and are a part of a framework designed under the assumption that it is fully in control of the execution loop and developed with different architectural styles to ensure their usefulness in specific contexts [9].

Therefore, the coordination model to represent interactions among Web services must be orthogonal regarding this heterogeneity. The *Generative Communication model*, alternatively Tuple Space Communication model, provides the illusion of a shared memory, called *Tuple Space*, to allow that the inter-process communication was uncoupled logically, temporarily, and spatially by means of *tuples*. A *tuple* is something like ["Gelernter",1989], where the components are supposed to be untyped, atomic values. The best-known example of coordination language based on Generative Communication is *Linda* [7]. Linda provides four basic operations: **eval** and **out** to create and insert new tuples into the tuple space; **in** to read and remove at the same time a tuple from the tuple space; and **rd** to read a tuple without removing it. Using these operators, sender and receiver processes cooperate among them in an uncoupled way. Linda is a model of process creation and coordination that is orthogonal to the base computation language in which it is embedded. It does not care how the multiple execution threads in a Linda program compute information; it deals only with how these threads have been created, and how they can be organized into a coherent program.

3 Description of the Web-Enabled Coordination Service

We have implemented a *Web-enabled Coordination Service* (WCS) to support the coordination among Web services. The coordination language Linda has been chosen to model the coordination functionality. The WCS is a network-accessible software that provides a set of services to communicate and to synchronize heterogeneous applications distributed over Internet. It offers highly uncoupled single or group communication services. These are data storage services that can hold them beyond the life of the generating distributed-applications and event notification services. Every distributed application, regardless of the hardware and operating system platforms where they are running, the programming language used to encode them and the middleware itself, must be able to gain access to these service to cooperate among them. The selected approach is through open Internet protocols, such as HTTP, and using standard-data format to encode exchanging data, such as XML.

As it is shown in the Fig. 1, WCS is composed by three software components that provide the previously presented services. Their roles and responsibilities into the coordination service are presented:

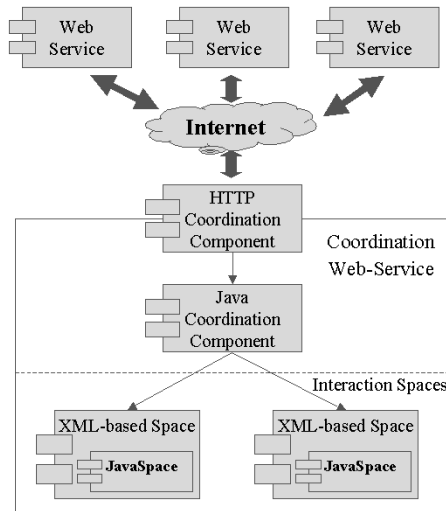


Fig. 1. Web-enabled Coordination Service

XML-based Space. This component has been developed based on JavaSpaces technology, and encapsulates inside an interaction space where a collection of processes through the interface of this component can cooperate among them exchanging XML documents. Besides, these XML documents can be defined in execution time. We have extended Linda with the notion of structured tuples which allows us representing XML documents as lists of attribute/value pairs. These XML documents are stored into the interaction space as Java objects.

The XML-based Spaces component tries to provide a more flexible way of working than JavaSpaces. It is because XML can be used to describe everything in a simple, powerful, and easy to understand way.

Java Coordination Component. This Java component is the core of the coordination service. It provides a collection of coordination operations, defined from the Generative Communication model previously presented, that are divided into two different interfaces called the *Basic Coordination* (BCI) and the *Reactive Coordination* (RCI) interfaces. The BCI offers a set of simple communication and synchronization operations among processes. These basic operations encourage a programming style where processes that invoke an operation can block until a communication is completed or a synchronization condition is achieved. This style could not be the most adequate to coordinate distributed processes. A reactive programming style based on distributed events can be used to design and build the coordination among distributed processes. To support this reactive style, the RCI provides operations so that a process can advertise its interest to generate a specific type of events, publish the advertised events and subscribe its interest to receive events of a specific type.

HTTP Coordination Component. This component plays a role as a Web accessible interface of the coordination component previously presented. It provides, through its two interfaces, called *HTTP Basic Coordination Interface* (HBCI) and *HTTP Reactive Coordination Interface* (HRCI), the same collection of operations than the Java Coordination Component. These interfaces describe coordination operations that are Web-accessible through Web protocols and data formats, such as HTTP and XML.

The nature of these interfaces hides the implementation details of the service so that it can be used for another Web-applications independently of the hardware and software platform where they are running and independently of the programming language in which they are written. This approach combines the best aspects of component-based development and the Web, and it is the cornerstone of the Web-Service-based model [6]. This model allows and encourages Web-Service-based applications to be loosely coupled, component-oriented and with cross-technology implementations.

4 Pattern Matching and Semantic Disambiguation

According to the LINDA model, the operation `in(x?)` tries to match the tuple `x?` with a tuple in the shared space. If there is a match, the tuple is extracted from the tuple space; otherwise, it blocks until a convenient tuple appears. The parameter for `in()` can be a *query tuple* with a wildcard, like in `["Gelernter", ???]`. The match is then *free* for the wildcard and *literal* for the constant values. Our proposal is based on this obvious observation: if this simple matching strategy is replaced with a *complex matching*, then very general kinds of interoperability can be achieved.

Considering this basic idea let us particularize the concept of *complex matching*. To this aim, we work with a version of Linda where the tuples admit a description by means of attribute/value pairs, like:

```
[(author, "Gelernter"), (year, 1989)].
```

Although this is still an untyped setting, this *bit* of structure allows us recovering information from a distributed context. Thus, if an operation `in()` is invoked in a different institution, where the term “author” is not used, but “creator” is used in its place, and if there is a convenient mapping between ontologies, then the request `[(creator, ???), (year, 1989)]` can be successfully satisfied. This type of *semantic matching* has already been exploited by our team in the context of GIS interoperability [8], and can be used to support multilingual interoperability. Interestingly enough, this semantic mechanism is implemented through Internet, using XML as transfer format.

Let us consider the following XML-data definition, extracted from [5].

```
Xml = (union String (cons Symbol (cons Att LXml)))
LXml = (listof Xml)
Att = ...
```

In that paper, this definition was implemented in Scheme, but it is clear how to directly translate it to languages as ML or Haskell, or, with a little more

effort, to any other programming language. Anyway, the important remark for our current presentation is that the “top level” structure of any XML document admits the expression

$$[(att1, <val\ 1>), \dots, (attN, <val\ N>)],$$

but where each $<val\ i>$ is structured (in particular, it can be XML-based). Now, Linda can be used in this XML context, providing a promising way to coordinate, communicate and collaborate on the net.

To achieve the semantic pattern matching we must face previously with the problem of word sense disambiguation. This problem is perhaps the greatest existing problem at the lexical level in natural language processing [11], and this skill is applicable to tasks such as information retrieval, machine translation, speech synthesis and pattern matching. The problem of disambiguation consists in determining which one of the senses of an ambiguous word is invoked in a particular context composed of a set of words related to the ambiguous word. A word is ambiguous (or polysemic) if its sense changes depending on the context.

There are different statistical approaches to solve this problem depending on the training material available. Supervised disambiguation methods are based on a previously disambiguated training set. Each occurrence of the ambiguous word is annotated with its contextually appropriated sense. Based on this information, the aim of these methods is to build a classifier, which correctly classifies new cases. On the other hand, unsupervised disambiguation methods try to distinguish among the senses of a polysemic word without help of disambiguated examples. Unsupervised methods are based only on the features that can be automatically extracted from unlabeled texts. There are also methods that use lexical resources such as machine-readable dictionaries, lexical knowledge bases or thesauri. These methods rely on the definition of senses in dictionaries and other lexical resources. Sometimes they are merged with training supervised or unsupervised methods.

The disambiguation method that we developed [8] can be considered as an unsupervised disambiguation method based on the hierarchical structure of WordNet and the notion of conceptual distance among concepts. The WordNet ontology is organized around the notion of *synset*, that is, set of synonyms that expresses a concept. By means of a voting system, our method takes a term from a thesaurus, where the terms are organized in hierarchical structures similar to trees whose nodes are the terms which maintain associations with their broader (ascendants) or narrower terms (descendants), and it tries to determine the “closest” sense (WordNet synset) to the senses of the other words in the whole branch that constitute its context.

As it has been previously mentioned, the XML-based Space component is based on JavaSpace Technology. It provides an object space for storing these XML-tuples, avoiding the development of a new XML-tuples space. However, the matching rules of JavaSpaces are not adequate for working over objects with structured fields. Objects are matched by complete fields, not within the contents of a field. This is owing to objects are serialized for storing them into the space, and the matching between objects is made applying an equality operator

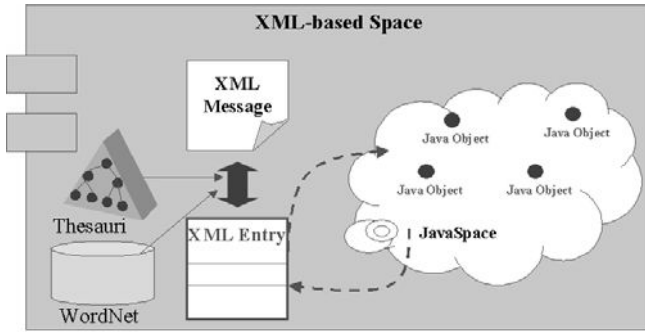


Fig. 2. Semantic resources

on the corresponding field value. In the case of a structured field, its value is the serialization of each component. The generic Java object that encodes any XML-tuple has structured fields to store the nodes of the XML tuple. This generic object has two structured fields to store the tag names and values of the XML-tuple. The tag name of the first node of the XML-tuple is stored in the first component of the tag-name field and its value in the first component of the tag-value field, and so on.

Therefore, it is necessary to increase the rules of JavaSpaces to allow the matching by the contents of a structured field. To resolve the rule restrictions, the matching is made in two steps. In the first step, the matching rules of JavaSpaces are used. The original template is saved for the second step and a copy of it is created for being used in the first step. The tag-value field of this copy is set to the null value (see our tentative of formalization in the next section). When it executes a read operation (provided by the JavaSpace interface) using this template object, a returned object represents an XML-tuple with the same XML-Schema as the template because the tag-name field is only considered for the matching. In a second step, it is invoked a particular matching method of the retrieved object, using the original template as a real parameter. The method checks that each not-null component of the template's value-field have the same value in the corresponding component of the retrieved object's value-field. If it returns a true value, the retrieved object matches the template according to the XML-tuple matching rules. Otherwise, the retrieved object has the same XML-Schema as the template but it does not match the template according to the second matching rule for XML-tuples. Then, the first step is made again until an object matches according to the XML-tuple matching rules.

If the words of the tag-name fields or of the tag-values fields have been extracted from a thesaurus, then it is possible, in the second step of the matching process (see Fig. 2), to make use of lexical and semantic resources, such as WordNet ontology and, by means of our disambiguating method, we can enrich the XML-based Space component of WCS with a semantic pattern matching where the retrieval of a XML document is based on the structural and semantic similarity of a document with a given template.

5 Towards a Formalization of Our Approach

When applying formal methods to model real-life systems, several degrees of abstraction can be used, depending on the objectives the analyst is looking for. In the *coordination* area, there are extremely abstract approaches as those of [2] or [3], where the formalisms for tuple-based coordination are based on process algebras. Taking into account that one of the main features of this Linda-like coordination is the associative access to a shared memory, the algebraic views in [2] or [3], where in particular any tuple space consideration is abstracted, could be considered too unrealistic. However, this has not to be observed as an inadequacy, but rather as a precise choice in order to analyze, in a way as simple as possible, some theoretical characteristics of the models (such as its expressiveness, for instance)

Other authors, as [10] or [12], having in mind different goals (as the analysis of event-based coordination), have presented formalisms in which tuple spaces are explicitly represented, and coordination gives rise to a lower level model (with respect to [2], [3]) by means of transition systems. However, in these approaches the matching process is completely abstracted through a *matching predicate*, considered as predetermined. It is quite clear that this approach is also too abstract to model our proposal.

In order to find the right *abstraction grain* for our task, three components appear as clearly distinguished. The first one is that our coordination model is constructed on top of another, more basic and predetermined, coordination artifact as JavaSpaces. The second one is that templates are structured and they must be handle in order to enable semantic XML-based interoperability. The third one is that the final coordination-service behaviour should be expressed in terms of the two previous points, and not in terms of some ad-hoc implementation details. Let us comment briefly each one of these three points.

In a first approach, we could replace the complex features of JavaSpaces by a formal model for Linda, for instance that of [12]. The main ingredients of this model (see [12] for details) are a set of tuples T ; a set of templates $Templ$; a matching predicate $mtc(templ, t)$, where $templ$ ranges over $Templ$ and t over T ; a choice operator $\mu(templ, \tau)$ extracting a tuple $t \in \tau$ which matches a (multi)set of tuples τ or returning an error element $\perp$ if no matching is available; and the standard Linda operations as *in*, *rd*, *out* and so on. Since this will be integrated as a *core* coordination model, auxiliary to define another one, let us denote each ingredient with the subindex *core*: mtc_{core} , μ_{core} , in_{core} , etc.

With respect to the complex matching process, it is necessary to give more (mathematical) structure to the templates set $Templ$. It is clear that templates are sorted with respect to generality; for instance, ["Gelernter", ???, ???] is more general than ["Gelernter", ???, 1988]. This relation can be axiomatized: $templ_1 < templ_2$ if $\forall t \in T_{core}, mtc_{core}(templ_2, t) \Rightarrow mtc_{core}(templ_1, t)$. Thus, $Templ$ is endowed with a partial order, with one minimal element for each arity of tuples; explicitly, the minimal elements are [????], [???, ???], ...

This scenario can be generalized if we assume that templates and tuples occur as sequences of pairs (**attribute,value**). Then, if wildcards are only allowed as values, other minimal templates appear:

[[author,???]],[title,???],[year,???]]

Let us introduce an operation¹ $min : Templ \rightarrow Templ$ associating to each template its minimal associated template.

In order to define the final coordination service in top of the two previous elements, at least two approaches are possible. In the first one, we considered not only the tuples in T_{core} , but also we consider other *virtual* tuples T_{virt} that can be constructed canonically from T_{core} by means of some disambiguation process (a typical example would be the case of a multilingual coordination service where, with the help of *dictionary mappings*, “virtual” tuples in other languages are considered). In this case, the new tuple space to be defined would be: $T_{def} := T_{core} \cup T_{virt}$. Then, the matching predicate mtc_{def} could be simply defined from mtc_{core} , but the choice operator, μ_{def} , would be in charge of constructing the possibly virtual counterpart of the tuple recovered by μ_{core} . The alternative is to define simply $T_{def} := T_{core}$, and then the burden of working with the semantic aspects is for both mtc_{def} and μ_{def} . In any case, a possible definition of the operation in_{def} could be sketched as follows. Giving a query $in_{def}(temp)$, the minimal template $min(temp)$ is used to query the underlying Linda Model as $rd_{core}(min(temp))$, in order to do no interference with other open processes. If $\perp$ is obtained (in other words, if the internal query blocks), this is the same for the original query (by definition of $min()$). Otherwise, a tuple is obtained and now can be processed in order to know (by means of mtc_{def}) whether it satisfies the original query; if it succeeds the tuple recovered by $rd_{core}(min(temp))$ is then removed.

Obviously, this very brief presentation is not only incomplete, but also let many points to be studied, in particular related to the soundness and fairness of the new defined coordination service. Nevertheless, we think that the main pieces are here established in order to analyze formally the theoretical properties of the Web coordination service proposed and, in particular, to prove if it accomplishes the very definition of some Linda-like coordination model.

6 Application Examples

We have showed the applicability of the WCS with a classical problem. We have implemented a variant of the classical concurrent programming problem of the Dining Philosophers (see Fig. 3). In our approach, several distributed philosophers, implemented in different programming languages (Lisp, Java, or HTML with JavaScript), cooperate on Internet (via HTTP and XML).

When a philosopher is inserted, Aristotle for example, his chopstick is introduced by the out operator:

<http://bubu.cps.unizar.es/CoordinationServlet?>

¹ This is an convolutive operator: $min(min(temp)) = min(temp)$.

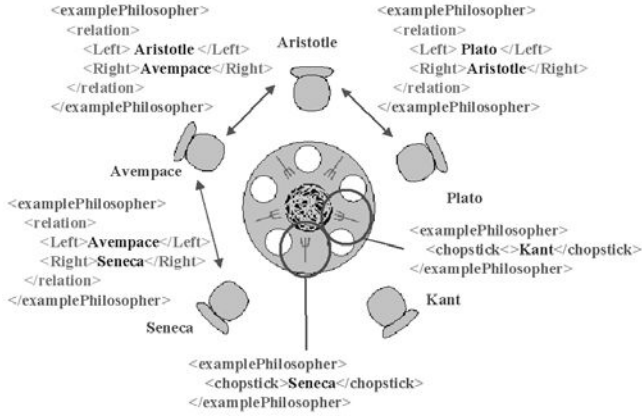


Fig. 3. The Dining Philosophers problem.

```
REQUEST=<?xmlversion="1.0"?> <CoordinationService>
  <function>out</function>
  <tuple>
    <examplePhilosopher>
      <chopstick>Aristotle</chopstick>
    </examplePhilosopher>
  </tuple>
</CoordinationService>
```

and it is also introduced the relation that represents who are on the right and on the left. It supposes to recover any neighborhood relationship in order to replace it by new relations when the new philosopher is introduced. The `in` operation with the tuple `[(Left, ???), (Right, ???)]` recover, for example, the relationship `[(Left, Plato), (Right, Avempace)]`. The recovered relationship is replaced by tuples representing that Aristotle is on the left of Avempace, and on the right of Plato. With these tools it is easy to describe the complete process.

Furthermore, the WCS model has been the conceptual base for the development in a real problem: the Location-based services (LBS) frameworks whose functionality may be integrated into end-applications through Internet, such as ERP or CRM systems [1]. LBS frameworks require the integration of Geographic Information services, location services and communication services. Required services are built according to the Web-service approach: their operations are provided through a standard, published interface to ensure interoperability, and are accessible via HTTP and XML.

7 Conclusion and Further Work

In this paper we have presented a Web-enabled Coordination Service to orchestrate heterogeneous applications based on the Generative Communication

model and implemented using Java and Internet technologies, such as HTTP and XML. It is an alternative to service-oriented architecture interaction model and independent of distributed object computing middleware. The coordination functionality of the service provides space and time uncoupling and represents an opportunistic strategy to use Web services. Furthermore, it integrates thesauri and ontologies to provide a semantic matching.

Open research issues are: (1) to work on dynamically discovering and intelligent chaining of services; (2) to extend the matcher to improve the semantic interoperability among Web services; (3) to incorporate daemons to Web services for supporting a reactive behaviour; and (4) to complete the formalization of our proposal.

References

1. P. Álvarez, J.A. Bañares, P.R. Muro-Medrano, and F.J. Zarazaga, *Integration of location based services for field support in CRM systems*, *GeoInformatics* **5** (2002), no. July/August, 36–39.
2. N. Busi, R. Gorrieri, G. Zavattaro, *On the expressiveness of Linda coordination primitives*, *Information and Computation* 156(1–2) (2000) 90–121.
3. N. Busi, R. Gorrieri, G. Zavattaro, *Process Calculi for Coordination: From Linda to JavaSpaces*, *Lecture Notes in Computer Science* 1816 (2000) 198–212.
4. N. Carriero, D. Gelernter, *Linda in context*, *Communications of the ACM* 32 (1989) 444–458.
5. M. Felleisen, *Developing Interactive Web Programs*, Summer School on Advanced Functional Programming, 2002. To appear in *Lecture Notes in Computer Science*.
6. R. T. Fielding and R. N. Taylor, *Principled design of the modern Web architecture*, *ACM Transactions on Internet Technology*, **2** (2002), no. 2 115–150.
7. D. Gelernter, *Generative communication in Linda*, *ACM Transactions on Programming Languages and Systems* **7** (1985), no. 1, 80–112.
8. E. Mata, J.A. Bañares, J. Gutiérrez, P.R. Muro-Medrano, J. Rubio, *Semantic disambiguation of thesaurus as a mechanism to facilitate multilingual and thematic interoperability of geographical information catalogues*, in *Proceedings 5th AGILE Conference on Geographic Information Science*, Universitat Illes Balears (2002) 61–66.
9. M. Mattsson, J. Bosch, E. Fayad, *Framework Integration. Problems, Causes, Solutions*, *Communications of the ACM* 42 (1999) no. 10, 81–87.
10. A. Omici, E. Denti, *From tuple spaces to tuple centres*, *Science of Computer Programming* 41 (2001) 277–294.
11. P. Resnik, D. Yarowsky *A perspective on word sense disambiguation methods and their evaluation*, in *ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What and How?*, Washington, D.C. (1997) 79–86.
12. M. Viroli, A. Ricci, *Tuple-Based Coordination Models in Event-Based Scenarios*, in *Proceedings of the IEEE 22nd International Conference on Distributed Computing Systems (ICDCS 2002 Workshops) – DEBS’02 International Workshop on Distributed Event-Based Systems – July 2002, Vienna, Austria, IEEE, 2002*

Mapping Nautilus Language into Java: Towards a Specification and Programming Environment for Distributed Systems[★]

Claudio Naoto Fuzitaki¹, Paulo Blauth Menezes¹, Júlio Pereira Machado¹, and Simone André da Costa²

¹ Instituto de Informática, UFRGS,
CEP 91501-970, Caixa Postal: 15064, Porto Alegre, Brazil
{fuzitaki,blauth,jhpm}@inf.ufrgs.br

² Centro de Ciências Exatas e Tecnológicas, UNISINOS,
Av. Unisinos 950, CEP 93022-000, São Leopoldo, Brazil
sac@exatas.unisinos.br

Abstract. This paper describes some of the features of Nautilus specification/ programming language that make it interesting to develop complex systems and then explains how to map a Nautilus construction into Java constructions. The Nautilus constructions presented are: actions (including nondeterminism and intra-action concurrency), aggregations and refinements.

1 Nautilus

Nautilus is originally object-based, textual and supports concurrent objects [1]. The language can be used to specify and/or program concurrent systems [2]. Some important extensions are classes [3], object restriction and a diagrammatic notation [2].

Its uncommon constructions (refinements, aggregation, etc) are based on its semantic domain: Nonsequential Automata [4]. This domain satisfies the diagonal compositional requirement, i.e. refinements compose (vertically), reflecting a stepwise description of systems, involving several levels of abstraction, and distributes through combinators (horizontally), meaning the refinement of a composite system is the composition of the refinement of its parts.

In this discussion of the language Nautilus we introduce some key words in order to help the understanding of the examples below. The specification of an object in Nautilus depends on if it is a simple object (whose construction does not depend on other objects) or a structured object resultant of an encapsulation, aggregation, refinement or parallel composition (which are constructions for building new objects over another objects). In any case, a specification has two

[★] This work is partially supported by: CNPq (Projects HoVer-CAM, Hyper Seed, GRAPHIT), CNPq/NSF (Project MEFIA) and FAPERGS (Project QaP-For) in Brazil.

main parts: interface and body. The interface declares the category (**category**) of some actions (**birth**, **death**, **request**). The body (**body**) declares the attributes (**slot**) and the methods of all actions (**act**). A birth or death action may occur at most one time (and determines the birth or the death of the object), much like constructors and destructors in object-oriented languages. An action may have enabling (**enb**) conditions and also alternative execution bodies (**alt**). An action may be a sequential (**seq...end seq**) or multiple (**cps...end cps**) composition of clauses. A multiple composition is a special composition of concurrent clauses based on Dijkstra's guarded commands where the valuation (**val**) clauses are evaluated before the results are assigned to the corresponding slots. Several objects are specified inside a unity (**spec...end spec**).

As discussed in [5], Nautilus may be used to teach concurrency in undergraduate courses, as the language allows students to develop complex concurrent systems abstracting away from low level details (such as semaphores, critical regions, etc). In Nautilus, concurrency can be specified in many levels: actions have internal concurrency, actions have concurrency in the same object, and objects are concurrent in the unity.

Several good properties and features of the language have been explored in previous works, confirming it is an elegant solution for concurrent and nondeterministic problems and for synchronization of complex concurrent systems:

- [2] compares Nautilus and UML [6], showing the expressive power of Nautilus as programming language and as system specification language;
- [7] shows it has anticipatory properties (anticipation in Nautilus is compositional and may be state-dependent, i.e., may depend dynamically on some conditions);
- [8] explored explicit and implicit nondeterminism present in refinement mechanism of the language.

2 From Nautilus to Java

Now we are looking for to mapping the Nautilus language constructions to construction of a well know language, Java [9], that was chosen because of previous experiences of one colaborator [10]. Presently the focus is a subset of the textual language that is object-based.

Nautilus should be useful for creating naturally concurrent solutions. This is an especially good propriety that could be better used if Nautilus were the first language of the programmers so we are making this mapping to create a Nautilus-Java translator that will be used in a course for first year computer science students.

Since there is no space to detail all the mapping only some of the main constructions mappings will be showed.

2.1 Actions

In Nautilus the executable units are not objects but actions (methods in Java). Objects are only a way to organize actions that are self-executable by default.

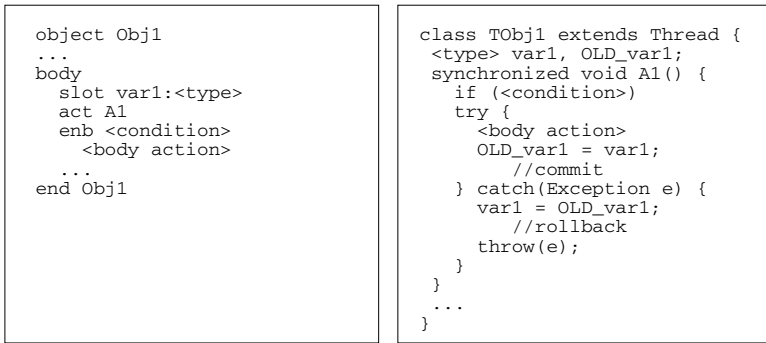


Fig. 1. A simple Nautilus action translated to Java

But the actions that will be executed are only the ones which are active, i.e., that do not have any kind of restriction (are not component of other actions, whose enabling condition is true, at least one alternative is active). So, if an action satisfies these conditions, besides the translation of its body, it is necessary to simulate the self-execution of the action inserting a (random) call in method **run()** of the correspondent Java object. Such procedure is necessary because the execution of enabled (active) actions is an internal nondeterminism.

Actions in Nautilus are transactional and in case of fail should rollback to the previous state (like in Database Management Systems). A simplified scheme to translate Nautilus actions into Java methods is shown in Fig. 1.

In this mapping any Nautilus slot generate two attributes inside the correspondent Java class. Slots in Nautilus are seen as object's attributes in object-oriented terms. One of them hold the old value of the slot for the case of fail. And the condition in **enb** is translated as a simple if clause. The condition is a simple boolean expression.

The scheme in the Fig. 1 is a simplification that works in the case when the action is not used by any composed action. When an action is composed by aggregation or refinement, its implementation will need to referenciate internal divisions of the component action, so the action will be translated into four methods (the action and its internal sections: body, commit and rollback). This case is shown in Fig. 2 (where the “_OLD” suffix was changed to a prime).

For the next constructions partly of the code necessary to deal with exceptions and auxiliary functions (commit, roolback) may be omitted for simplicity.

2.2 Intra-action Concurrency

Concurrency inside an action is expressed by keyword **cps** used to make concurrent assigns, like in Fig. 3. It could be easily translated to Java using temporary variables and sequential execution.

For system with multiples CPUs it is possible to create a thread to evaluate each right expression, but to optimize this behaviour it would be necessary to verify how complex an expression need to be for compensate the extra processing of creating threads.

<pre> object Obj2 ... body slot S1,S2:<type> act A1 <body action> // alters slots S1 and S2 ... end Obj2 </pre>	<pre> class TObj2 extends Thread { <type> S1, S1', S2, S2'; synchronized void A1() { try { A1_body(); A1_commit(); } catch(Exception e) { A1_rollback(); } } void A1_body() { <body action> } void A1_commit() { S1' = S1; S2' = S2; } void A1_rollback() { S1 = S1'; S2 = S2'; } ... } </pre>

Fig. 2. Nautilus action translated to Java

<pre> slot S1, S2:<type> act A1 cps val S1 << <expression1> val S2 << <expression2> end cps </pre>	<pre> <type> S1, S1', S2, S2', Temp; void A1_body() { Temp = <expression1>; S2 = <expression2>; S1 = Temp; } </pre>

Fig. 3. Intra-action concurrency translation

2.3 Nondeterminism

In Nautilus, an action may have alternative execution bodies (introduced by the keyword **alt**). To translate such nondeterminism, initially it was thought of using a choice operator that could be the built-in **random()** function as in Fig. 4.

But it was perceived that this would not work in case of actions that are restrict by **request** or are used as a refinement component (in this cases the formal semantics demands that if there are alternatives that fail and others that succeed, the chosen one should be one that succeeds), so it was necessary to use another pattern where alternatives are tried until one of them succeeds. For example, an action **A** with alternatives **A1**, ..., **An** should be translated like shown in Fig. 5.

2.4 Action with Parameters

In case of actions with parameters (like in an aggregation) there are more complexities to consider. Parameters impose restrictions in the execution order of actions. The initial and final parts of procedures become synchronization points where the procedures receive and send actual parameters.

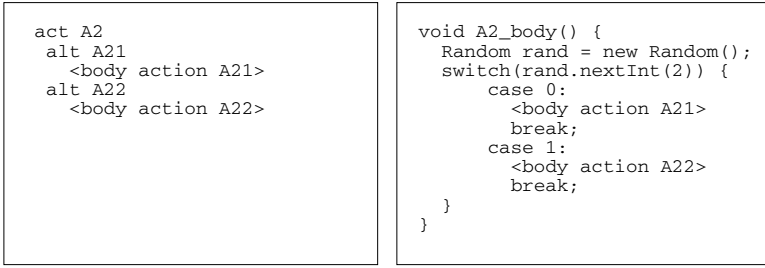


Fig. 4. Initial Nondeterminism translation

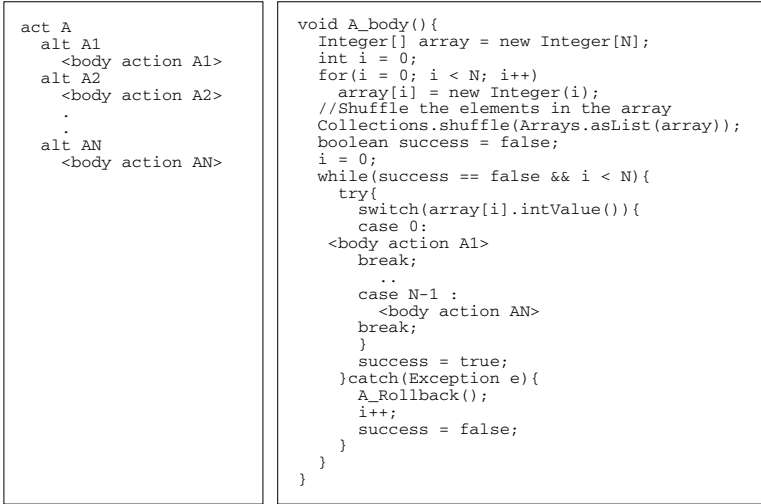


Fig. 5. Nondeterminism translation

Every action that receives or returns parameters is implicitly of the type **request**, since it cannot self-execute alone. Parameters references are by identifiers, not by position and are possible only to exported actions (actions which are public to other objects).

A illustrative example is the program in Fig. 6. In this example, objects **Obj1** and **Obj2** have actions with parameters, and a third object **ObjAg** is an aggregation acting as synchronization of input/output of the first objects.

Depending of the body of **A1** and **A2**, there are the following possibilities:

- Sequential Execution

As shown in Fig. 7, one of the actions supply the output parameter in the first instruction through the keyword **agg**, that is the input parameter of the second action, that now can execute and returns an output parameter that is used by the first action.

- Parallel execution

When both actions supply parameters in the beginning, both can be executed. See Fig. 8.

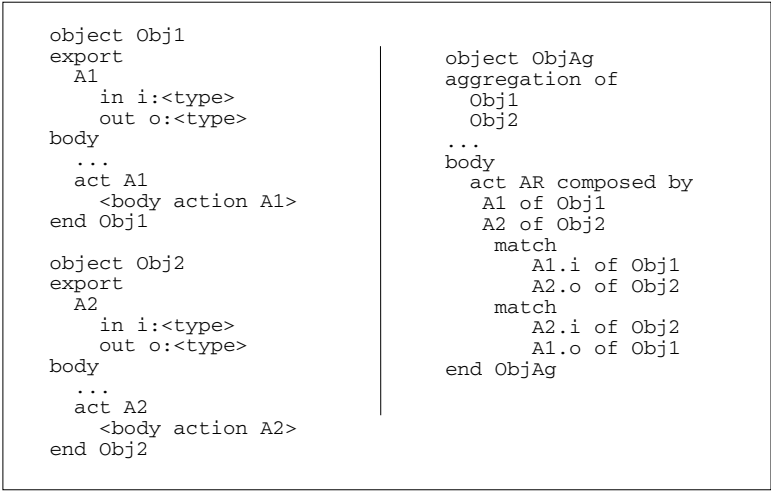


Fig. 6. Parameter example

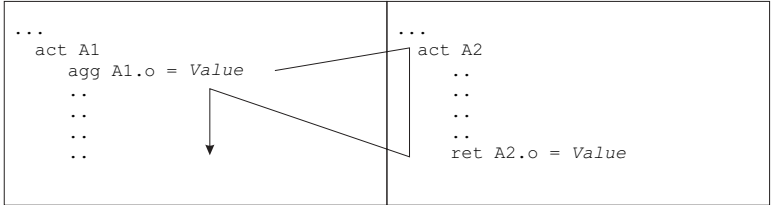


Fig. 7. Sequential execution

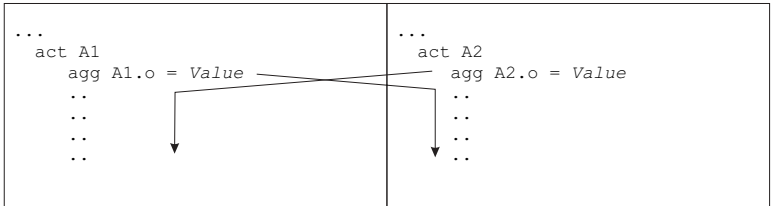


Fig. 8. Parallel execution

- No execution
The no execution case means that actions are mutually dependent and do not occur (Fig. 9). The semantic is empty, for all effects is like they have **enb false** clauses.

Based in these cases it is necessary to make an analysis: Which parameters are provided by **agg** clauses? If no action can start just with these parameters, then it is a no execution case, else choose any action that can be executed.

The Java translation of Nautilus parameters can be done using “pseudo-slots”. In this way each parameter would be a slot like “ActionId.Param.Id’TypeId”, with the restriction that cannot be used out of the

<pre> ... act A1 ret A1.o = Value </pre>	<pre> ... act A2 ret A2.o = Value </pre>

Fig. 9. No execution

<pre> object ObjBase category birth Start body ... act Start <body action Start> act Action1 <body action Action1> act Action2 <body action Action2> act Action3 <body action Action3> ... end ObjBase object ObjRef over ObjBase category birth Start body act Start Start act ActionComp1 seq Action1 Action2 end seq end ObjRef </pre>	<pre> class TObjBase extends Thread { ... TObjBase(){ super(); Start(); } void Start(){...} void Action1(){...} void Action2(){...} void Action3(){...} ... } class TObjRef extends Thread { TObjBase ObjBase; TObjRef(){ super(); Start(); } void Start(){ ObjBase = new TObjBase(); } void ActionComp1() { try { ActionComp1_body(); ActionComp1_commit(); } catch(Exception e) { ActionComp1_rollback(); } } void ActionComp1_body(){ ObjBase.Action1_body(); ObjBase.Action2_body(); } void ActionComp1_commit(){ ObjBase.Action1_commit(); ObjBase.Action2_commit(); } void ActionComp1_rollback(){ ObjBase.Action1_rollback(); ObjBase.Action2_rollback(); } public void run(){ boolean the_end = false; Random rand = new Random(); while (!the_end) switch(rand.nextInt(1)) { case 0: ActionComp1(); break; } } } </pre>

Fig. 10. Refinement

correspondent action “ActionId” and that attribution is done only in **agg** and **ret** sections.

<pre> object Tic category birth Start export WriteTic body ... act Start <body action Start> act WriteTic <body action WriteTic> end Tic object Tac category birth Start export WriteTac body ... act Start <body action Start> act WriteTac <body action WriteTac> end Tac object TicTac aggregation of Tic Tac category birth Start body act Start composed by Start of Tic Start of Tac act WriteTicTac composed by WriteTic of Tic WriteTac of Tac end TicTac </pre>	<pre> class TTic{ ... TTic(){ Start(); } void Start(){...} void WriteTic(){...} ... } class TTac{ ... TTac(){ Start(); } void Start(){...} void WriteTac(){...} ... } class TTicTac extends Thread { TTic tic; TTac tac; TTicTac() { super(); Start(); } void Start() { tic = new TTic(); tac = new TTac(); } void WriteTicTac() { Tic.WriteTic(); Tac.WriteTac(); } public void run() { boolean the_end = false; Random rand = new Random(); while (!the_end) switch(rand.nextInt(1)) { case 0: WriteTicTac(); break; } } } </pre>

Fig. 11. Aggregation

2.5 Refinement

Refinement is an implementation of an object over another. An action of a refinement object is constructed over the actions of a base object. Its the only construction that imposes an order, respecting sequencially, concurrency and nondeterminism. It can be translated into Java as method calls with a transaction restriction (Fig. 10).

2.6 Aggregation

Its is one of the main composition mechanisms of Nautilus. Each object can be viewed as a system and the aggregation establishes some synchronization points between the aggregated objects. A single example is shown in Fig. 11.

In this case aggregation takes off the possibility of an action to self execute, but if, for example **Tac** have an action called **WriteTac2** and no reference to this action in the object **TicTac**, this action could self execute, but it would be inaccessible (the aggregation would hide it). And the **switch** of the **run()** method would have a **case 1: Tac.WriteTac2()**.

In this simple example there is no imposed sequentiality of how the component actions would have to execute, but in case of aggregation using actions with parameters there are more complexities as showed above in section 2.4 .

3 Conclusion and Future Works

This paper described features of Nautilus language and showed how some core constructions of Nautilus (intra-action concurrency, nondeterminism, aggregation, refinement) can be mapped into Java. We plan to continue the mapping to include others construction like classes. This shows that is feasible to implement higher level concurrent Nautilus constructions in Java and is a first step to make an unified specification and programming environment for distributed systems.

References

1. Menezes, P.B., Sernadas, A., Costa, J.F.: Nonsequential automata semantics for a concurrent, object-based language. In Cleaveland, R., Mislove, M., Mulry, P., eds.: Proc. of the 1st US – Brazil Joint Workshop on the Formal Foundations of Software Systems. Volume 14 of Electronic Notes in Theoretical Computer Science., New Orleans, EUA, Amsterdam, Elsevier (2000)
2. D’Andrea, F., Menezes, P.B., Fuzitaki, C., Iio Machado, J., Costa, S.: Nautilus, a diagrammatic specification and programming language. In: PDCS2002 - 14th International Conference on Parallel and Distributed Computing and Systems, Cambridge, USA (2002)
3. Carneiro, C.: Identified non sequential automaton as support for classes in nautilus (in portuguese). Master’s thesis, PPGC da UFRGS, Porto Alegre (1999)
4. Menezes, P.B., Costa, J.F., Sernadas, A.: Refinement mapping for (discrete event) system theory. In: Proceedings of the Fifth International Conference on Computer Aided System Technology, EUROCAST 95. Number 1030 in Lecture Notes in Computer Science, Springer-Verlag (1996) 103–116
5. Carneiro, C., Veit, T., D’Andrea, F., Menezes, P.B.: Nautilus: its concurrent and distributed characteristics as an academic language. In Arabnia, H.R., ed.: Proc. of the International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, EUA, Athens, C.S.R.E.A. (1999) 1919–1925
6. Fowler, M., Scott, K.: UML Distilled: A Brief Guide to the Standard Object Modeling Language (2nd Edition). Second edn. Addison-Wesley (2000)
7. Menezes, P.B., Costa, S.A., Machado, J.P.: Nautilus: A concurrent anticipatory programming language. In: Conference of Computing Anticipatory Systems, CASYS 2001. (2001)
8. Menezes, P.B., Machado, J.P., Costa, S.A.: Explicit and implicit nondeterministic refinement for concurrent, interacting systems. In: The International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA’02, Las Vegas, USA (2002)

9. Deitel, H.M., Deitel, P.J.: Java : how to program. 3rd edn. Upper Saddle River : Prentice Hall (1999)
10. Barbosa, J.L.V., Du Bois, A., Pavan, A., Geyer, C.F.R.: Holojava: Translating a distributed multiparadigm language into java. In: CONFERÊNCIA LATINOAMERICANA DE INFORMÁTICA, 27., 2001, Mérida, Venezuela (2001)

Design of a Medical Application Using XML Based Data Interchange*

C. Mariño, C. Abalde, M.G. Penedo, and M. Penas

LFCIA Lab, University of A Coruña
Campus de Elviña s/n, A Coruña, Spain
castormp@fi.udc.es
{carlos,penedo,infmpc00}@dc.fi.udc.es
<http://www.lfcia.org>

Abstract. The recent proliferation of computing devices and the contexts in which they are used demand diversity in distributed applications as well. The objective of our research is the development of a medical framework where information from patients can be accessed from heterogeneous and (possibly) mobile computing environments. Moreover, high availability and reliability are also milestones in that system. The former objective is achieved by using eXtensible Markup Language (XML) for the communication medium, in combination with eXtensible Stylesheet Language (XSL) transformations to allow different kinds of clients access the data. High availability is achieved by using a concurrent and distributed language, Erlang/OTP, for the development on the server side. Also, in the server side, techniques coming from the formal methods area are applied to improve the system design and performance and to ensure the system correctness. And finally, reliability, confidentiality and authentication, fundamental items in the data communications, are accomplished by mean of the Secure Socket Layer (SSL) protocol.

1 Introduction

Traditional medical applications are stand-alone programs which does not allow for the sharing of the data from the patients in a comfortable way. Clinicians can not get the data from the patients if they are outside the hospital, and in an emergency that could not be the case. Moreover, devices like PDAs connected over wireless channels or laptops connected by fixed networks could be linked to the hospital's database, and so provide the data when necessary, independently of the location of the patient or of the clinician. To fulfill all those requirements, a framework must be designed to allow the communication between the server system and the (possibly) mobile device handled by the clinician. The heterogeneity of computing platforms manifests itself in CPU speed, memory, display capabilities and network bandwidth. Of these, displays will likely take the most diverse forms as already visible in the latest developments. Examples are mobile

* Partially supported by MCyT Project TIC 2002-02859 and Xunta de Galicia Project PGIDT01PXI10502PR.

phones, Personal Digital Assistants (PDAs) displays or workstation displays. The former are invariably more limited in size and quality than the latter. To solve these problems and achieve client-independence the emerging technology based on the couple XML/XSL [1][2][3][4] seems to be the best option today. The mobility of clients further complicates information processing, retrieval, and delivery in the mobile web. Conventional client/server model on the web is often based on *Remote Procedure Call* (RPC) or *Remote Method Invocation* (RMI). Here XML-RPC [5] has been employed to invoke procedure execution in the server.

Another typical requirement from distributed systems are reliable communications. Patients' data must be confidential, so that data can only be accessed from authenticated clients. Moreover other issues like integrity (message contents cannot be altered by unauthorized entities) and authentication (non one can disguise oneself as the legitimate communication party) must be carefully considered. One way to address these problems is to employ *secure HTTP*, namely HTTP over Security Security Layer (SSL) [6]. SSL, an open, non-proprietary protocol is perhaps the most common way of providing encrypted transmission of data between clients and HTTP servers (HTTP-S is the runner up). Built upon private key encryption technology, SSL provides data encryption, server authentication, message integrity, and client authentication for any TCP/IP connection.

In this paper, a distributed client-server framework which allow for clinician-patient's data interaction is outlined. From the design stage, where design patterns are the fundamental tool, to the implementation (server, communication channels and clients) are described. The main goal of this work is to build a robust system which allows for remote querying the hospitals' databases, including the execution of some medical algorithms implemented in our laboratory [7][8] through the XML-RPC protocol.

This work is organized as follows: in section 2 an overview of the architecture of the system is presented, focusing in the objectives of the framework in the application domain, objectives of the chosen architecture, design patterns, client-server composition, etc. In section 3 different studied solutions for the client design and implementation are commented, and final solution is described. Communication's technology is described in section 4, while the server side is explained in section 5. Finally, conclusions and future work are commented in section 6.

2 Architecture Overview

Our system employs a conventional client-server architecture [9], as shown in figure 1. to accomplish the independence from the client technology, wrapping and to set a highly distributed and scalable system, a 4-tier system was considered as a good option option. Figure 2 shows an overview of such system.

Tier 1, data access layer: in this layer requests to the database are attended and data retrieved is sent to the application layer.

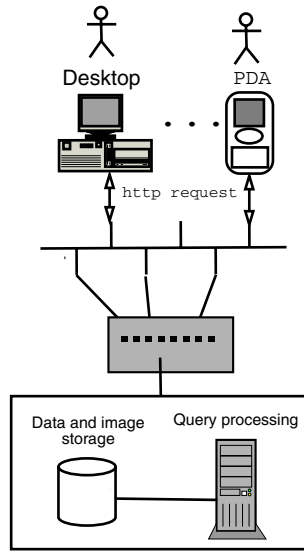


Fig. 1. Schematic representation of the proposed system

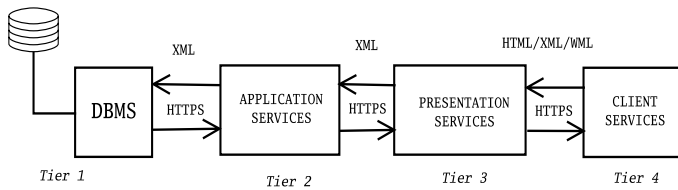


Fig. 2. The system has been designed as a 4-tiered architecture

Tier 2, application layer: the server executes the requested program (querying the database if necessary) and sends the results to the presentation layer. Requests come in the way of XML-RPC calling, and data are sent to the presentation layer in the form of XML formatted strings.

Tier 3, presentation layer: this layer allows the server to perform the necessary data format conversions to allow a correct understanding between the client and the server.

Tier 4, client layer: it is in this layer where (heterogeneous) clients perform their requests and present the results. Depending on the kind the client belongs to, that requests can be sent as an XML/HTML/WML string, and the results will be served (by the presentation layer) in the same format.

The more important tool employed for designing the system has been design patterns [10]. For example an adapter pattern was employed for the translation of the distinct client-formated queries (which could be a HTML/WML/XML flavored request) to the XML format required by the server, a chain of respon-



Fig. 3. Screen from the client program-application with a patient image loaded

sibility pattern to allow the existence of several applications in the server side having the same interface (as will be shown), or the strategy pattern, which encapsulates the several methods to compute the medical parameters required by the clinicians. But also distributed patterns appear in the design [11]: single threaded execution pattern is needed to allow only a client update the data of a given patient at a time, the scheduler pattern allows the execution of the methods given an execution policy (in our case a very simple one, an static priority is assigned to each of the medical applications, based on the required execution time, which is already known), and the read-write lock pattern, which avoids unnecessary waiting to read data of a patient by allowing concurrent reads by allowing only single threaded access when it is being updated.

3 Client Design

A client initiates a connection via a browser, and if authentication is successful, data from patient can be retrieved (figure 3). The client front-end's is an applet which is downloaded from the server when the connection is established. The graphical user interface for the interaction with image data (like the depicted in figure 3) is obtained by means of the Java-Plug-in which comes with Java 2 SDK, Standard Edition, Version 1.3.1 [12].

The overall operation flow at the client-side can be described as follow:

- Once the interactive applet has been downloaded by the client, a clinicial makes HTTP request to web server first. The web server will handle the request and generate the XML documents to the clinician. The XML documents include images, personal data, and supplement information (as can be seen in figure 4).
- Once the client receives the HTTP response, the XML document for supplement information will be parsed at the client side by the Java Applet.
- Upon request, procedure execution will be invoked (using XML-RPC) based on the data sent by the client. That data could be points picked in the image by the clinician, a set of images to register, etc.
- Once the procedure ends, the related information is sent to the client and displayed in the browser window.

```

<med>
  <paciente ssN="666">
    <nombre>"Nombre del paciente"</nombre>
    <fnacim>"1/1/2001"</fnacim>
    <direccion>"1,calle sin numero, Valencia"</direccion>
    <imagenpaciente>
      <fadquicsicion>"3/2/2002"</fadquisicion>
      <imagenId>"732819"</imagenId>
      <coordNervioOptico>
        <x>345</x>
        <y>234</y>
      </coordNervioOptico>
      <radio>100</radio>
    </imagenpaciente>
  </paciente>
</med>

```

Fig. 4. XML-RPC response which provides the client with the data from the patient

4 Access Technology

To assure confidentiality and reliability in the communications, the most widely used secure Web encryption has been employed: the Secure HyperText Transport Protocol (HTTPS) on the Secure Socket Layer (SSL) version 3.0 with 40 and 128 bit RC4 encryption. The SSL [13] protocol has become most widely used method for encrypting and authenticating communications.

Our approach to deal with the heterogeneity of the clients focuses on documents or data and transforms that visualizes the data to fit the computing capabilities of different devices, while preserving semantics of data. This division of the underlying data and the way it gets displayed mimics the Model-View-Controller (MVC), a well known and frequently used design pattern to develop interactive applications with flexible human-computer interfaces.

XML has been chosen as the medium to represent the data the clinician gets as well as the instructions for calling the algorithms which process the data. XML is a markup language for documents containing structured information. Once the data is written in XML, as associated XSL document can be written to define the way it gets displayed. The XML document is platform-independent and thus corresponds to the *model*, whereas the style sheet depends on the displaying device, and corresponds to the *view*.

MVC separation of view and model offers the following advantages over conventional application-centric environments:

- Model-view separation at the document level via XML/XSL documents.
- Simple communication protocol with a standard message format based on (encrypted) ASCII XML messages.

Finally, as commented above, using XML-RPC [5], procedure calls are wrapped in XML establishing that way a simple pathway for calling functions.

XML-RPC is a quick-and-easy way to make procedure calls over the Internet. It converts the procedure call into XML document, sends it to a remote server using HTTP, and gets back the response as XML. Other considered (although finally discarded) protocols where SOAP and IIOP (CORBA), both of them popular protocols for writing distributed, object-oriented applications, well-supported by many vendors, although they are much more complex than XML-RPC, and require fairly sophisticated clients. In addition, XML-RPC has fewer interoperability problems than them.

5 Server Design

Having in mind the global system architecture overview and the goals and features that we were looking for, in this section we give a brief introduction to the main options of the design and implementation on the server side of this medical environment.

The key features of the server are the following:

- Like all the critical and well designed applications, the operations over the server must satisfy the *ACID properties*: Atomicity, Consistency, Isolation and Durability.
- The server must be executable in a *cluster of low cost computers* and must be *scalable*. A new set of nodes can be added to the cluster without changing the system implementation and without stopping it. The system should be able to satisfy the requirements in a simple context (downwards scalability); and also should be scalable to allow in an easy way an enhancement of the performance by adding new resources to the architecture, serving more concurrent users (upwards scalability).
- *Fault tolerant*, meaning that if one of the nodes of the server system architecture crashes, the users are not going to notice any problem with the service. Whenever this happens, another node is going to assume the role of the crashed in a transparent way. With 24x7 expected uptime, there is a need for some kind of mechanisms to keep working at least in a degraded mode when some kind of fail happens.
- *Integration*: the server must provide some mechanisms to integrate it with other previous applications that we want to keep working.
- *Secure*. The server must be secure, limiting the functionality which can be used by each user.
- *Low cost*. The goal is to satisfy all the above requirements with an architecture that can reduce the implementation and deployment costs.

5.1 Some Technological Approaches

The requirements imposed by the server access technology and by the user interface and access devices have been taken into account when the server implementation technology was chosen. In this section we give a brief introduction to the main considered and finally discarded technologies. In section 5.2 the finally chosen technological approach is presented.

J2EE Based Server. J2EE (Java 2 Enterprise Edition) is an standard framework widely used in the industrial world. There are a lot of commercial and free implementations, that can be used for building enterprise applications (scalable, transactional and secure applications) with Java. J2EE provides, among other things, the following APIs: JDBC (relational database access layer), XML APIs, servlets+JSP+JSTL (view layer in web applications) and EJB (distributed, transactional, secure and scalable components that wrapper business logic).

Nowadays, J2EE is the most popular technological approach. The main reasons that explain this popularity are:

- The popularity and characteristics of the underlying programming language: Java.
- The great portability of the J2EE based products.
- The high number of available implementations of the J2EE APIs, some of the Open Source.

.NET Based Server. .NET is a framework comparable with J2EE, but with two main differences: it's a less mature product and it's supported by only one vendor (Microsoft). Microsoft .NET defines a CLR (Common Language Runtime) and an IL (Interface Language). All the .NET compatible programming languages must provide an IL output. .NET also provides a set of technologies symmetrical to the provided by J2EE: ADO.NET vs JDBC, ASP.NET vs JSP and COM+ vs EJB.

Integration Technologies: CORBA. CORBA [14][15] is an standard framework widely used in the industrial world. There are a lot of commercial and free implementations, all of them ready to build distributed object oriented applications that can run in very heterogeneous environments. CORBA let us to develop distributed applications (mainly in intranets) using remote object invocations techniques, which do not care of the programming language of the calling object or of the called procedure which implements a method from an object, the platform (operating system and hardware) or the intermediate communication networks.

There are CORBA implementations to almost all operating systems and most common programming languages. CORBA uses a binary communication protocol (IIOP) and it's standardized by the OMG since 1995. It's a very well suited technology to the integration of applications written in different languages in intranets, but not in Internet:

- The use of IIOP gives a lot of problems when the working network contains firewalls.
- CORBA is not supported by all the vendors: Microsoft doesn't support it.

Integration Technologies: Web Services. Web Services are the newest integration technology of heterogeneous applications. It uses XML over HTTP

(SOAP, Simple Object Access Protocol) as the format to interchange data between the integrated applications. Web Services have been adopted very fast by the industry, mainly because it can be used in the domains where CORBA doesn't fit well (integration over Internet). Then, CORBA and Web Services are complementary integration technologies.

There are Web Services APIs to J2EE, .NET or LAMP implemented by many vendors. Additionally, Web Services is very well suited to the integration of applications in Internet: the communication protocol, SOAP, wraps the request over HTTP (or SMTP), removing the problems with firewalls.

Other. Apart from the previous solutions, there are more choices to the design an implementation of the server side: from platforms similar to .NET or J2EE, like LAMP (Linux, Apache, MySQL and Perl/PHP/Python), to systems based on technologies like PVM or MPI or the classical Java RMI or DCOM. In the following section we introduce the final selected technology, which represents an original approach that could be combined with some of the previous described technologies.

5.2 Erlang/OTP Based Server

The concurrent and distributed language Erlang/OTP [16] has been chosen for programming the system server side. Erlang/OTP is a functional language originally developed by Ericsson, that has suitable features for the implementation of soft real-time fault-tolerant distributed processing systems. For example, some development tools, some libraries for the creation of graphical user interfaces, and even a distributed database. Currently, many companies like Alteon, (now part of Nortel Networks) Cellpoint, Corelatus, or One2One (now part of T-Mobile) use it. And, of course, many Ericsson products like the AXD301 and ANx are developed in whole or part in Erlang/OTP.

In Erlang/OTP, the combination of the functional paradigm and the parallel computation defines a declarative language, with almost no side effects and with a high level of expressiveness, abstraction and ease for prototyping.

Erlang/OTP is specially suited for real time software, and for distributed and fault tolerant systems. All of these are features of our target system. The language is based on asynchronous message passing, transparent communication of values, high order communications, and it is designed to support a high amount of concurrent processes. The language is suited for distributed systems development, and allows transparent allocation of processes among different nodes. Besides, Erlang/OTP includes primitive functions for supporting fault tolerance and allows code replacement without having to stop the system.

Integration. The use of Erlang/OTP as the implementation language doesn't limit the integration of the server with any previous developed application or system. Erlang/OTP provides mechanisms to connect and integrate the system using mainly two approaches:

- The Port mechanism, which allows to integrate in the system sections implemented in Java or C.
- The ORB (Object Request Broker) library, which allows the integration with other systems using CORBA technology.

Beowulf Cluster. Erlang/OTP seems a very well suited language to the implementation of our system. Additionally, because of its distributed message passing nature, it complements very well with the use of a low cost Beowulf cluster architecture, which is basic to reach the high scalability and availability goals.

Additionally, the use of a Beowulf cluster gives us other advantages like the existence of an OpenSource community with experience in high speed networks, distributed systems and clustering, the availability of the source code and development tools, the ease for migrating to other Unix versions, and the support for several platforms.

Past experiences. The selection of Erlang/OTP as the implementation language of the system is based on the set of characteristics that have been presented in this section. But, moreover, there are other reasons:

- The previous experience of the group on the development of other distributed and high availability systems based on Erlang/OTP, like a video on demand server [17] (VoDKA, Video on Demand Kernel Architecture) or a risk management information system, with an architecture similar to the proposed in this paper.
- The use of Erlang/OTP allows the application of techniques coming from the formal methods area in order to improve the system design and performance [18], and to ensure (at least partially) the system correctness [19]. These techniques can be used more naturally when taking as input a high level declarative language as Erlang/OTP, and their results are specially appreciated in critical applications such as a medical system. In section 5.3 we introduce the main concepts of this area.

5.3 Software Verification

Any software product must be written using some programming language. Some languages families reduce the number of mistakes introduced by the programmer. Declarative languages (logic or functional), like Erlang/OTP, are a step forward in the reduction of mistakes of software. The fact of work with a high level declarative language, with a clear separation between system logic and control details, allows programmers to focus in the problem. It has been proved that the number of lines of code in a system developed using Erlang/OTP can be more or less half the same as the same system developed in C++ [20].

The application of formal methods to the verification of information systems is a growing area. In the last years, this kind of methods have been employed to the verification of hardware [21] and communication protocols [22]. Its application to the software engineering area is slower because of the high complexity

of information systems. There is still a lot of work to be done in this area, but due to the advantages of using Erlang/OTP, the contribution to the software verification area using this language are very interesting.

Verification of Erlang/OTP Systems. The two most common techniques to verify a software system are the exploration of the system's state diagram (*model checking*) and the *theorem provers*. The first method it's more automatic, but it can't work with infinite structures and has a lot of problems with the state explosion. The second method it's more powerful and can work with infinite structures, but it's more complex and needs human participation during the proving process.

In both cases, the verification objective is to check that the system fulfill a set of properties, coded using some kind of formal logic (μ -Calculo, CTL, LTL, XTL, etc.). Nowadays, the trend it's to merge both techniques.

The Erlang/OTP verification initiative has two main branches:

- The development of EVT: a verification tool based on then theorem prover technique. With this tool the user can reason about Erlang/OTP programs at a very high level, trying to prove properties or behaviors encoded using μ -Calculo.
- The study of the model checking technique: Erlang/OTP programs are translated to μ CRL (a process algebra), then, from this representation it's obtained the state diagram and finally a model checker (Caesar/Aldebaran, CADP) is used to verify some properties encoded with μ -Calculo over the state diagram.

6 Conclusions and Future Work

In this paper a medical application for the access to patients databases has been presented. This approach allows clients with different computing capabilities to access the data from patients. XML, which servers as the communication medium, is a standard that has already gained wide acceptance and provides a powerful medium for data exchange, visualization specification and procedure execution by means of the XML-RPC method invocation procedure. Moreover all the communications take place over secure channels, through to the employment of encryption with HTTPS on the SLL version 3.0. The concurrent and distributed language Erlang/OTP [16] has been chosen for programming the system server side, providing a robust environment for the execution of the server procedures. The utilization of Erlang as the programming language of the server allows for a better verification of the more important parts of the system by means of formal methods of verification.

Currently, system is still under development, and many tasks as robustness assessment, performance verification, formal verification of whole the server-side procedures, so as testing in a real clinical environment must be performed.

References

1. Extensible markup language:<http://www.w3.org/xml>. See also <http://www.xml.com/>.
2. David Megginson. *Structuring XML Documents*. The Definitive XML Series from Charles F. Goldfarb. Prentice-Hall, 1998.
3. T. Bray, J. Paoli, C. M. Sperberg-McQueen, and E. Maler (Eds). "Extensible Markup Language (XML) 1.0 (2nd Edition)". W3C Recommendation, 2000. URL:<http://www.w3.org/TR/1999/REC-xml-19980210>.
4. J. Clark. "XSL Transformations (XSLT) Specification Version 1.0", 1999. URL:<http://www.w3.org/TR/1999/WD-xslt-19990121>.
5. Simon St. Laurent, Joe Johnston, and Edd Dumbill. *Programming Web Services with XML-RPC*. O'Reilly, 2001.
6. A. O. Freier, P. Kariton, and P. C. Kocher. The SSL protocol: Version 3.0. Technical report, Internet draft, 1996. Will be eventually replaced by TLS.
7. F. Pardo, V. Leborán, C. Mariño, M.G. Penedo, M.J. Carreira, A. Mosquera, D. Cabello, F. Gómez Ulla, and F. González. Retinal angiography image registration applied to hemodynamic variable measurement. In *Proceedings of the IX Spanish Symposium on Pattern Recognition and Image Analysis*, volume II, pages 139–144, 2001.
8. A. Mosquera, R. Dosil, V. Leborn, F. Pardo, F. Gomez-Ulla, B. Hayik, A. Pose, and M. Rodriguez. Art-vena: Retinal vascular caliber measurement. In *1st Iberian Conference on Pattern Recognition and Image Analysis (IBPRIA' 2003)*, Mallorca, Spain, June 2003. publication pending.
9. Robert Orfali, Dan Harkey, and Jeri Edwards. *Client/Server Survival Guide*. John Wiley & sons, 3rd edition, 1999.
10. Erich Gamma, Richard Helm, Ralph Jonson, and John Vlissides. *Design Patterns, Elements of Reusable Object-Oriented Software*. Professional Computing Series. Addison-Wesley, 1995.
11. Mark Grand. *Patterns in Java: a catalog of reusable design patterns illustrated with UML*, volume 1. New York, John Wiley & sons, 1998-1999.
12. Java 2 sdk, standard edition version 1.3.1, plug-in installation notes. <http://java.sun.com/j2se/1.3/install-linux-sdk.html>.
13. K.E.B. Hickman. The SSL protocol. <http://www.netscape.com/newsref/ssl.html>, December 1995.
14. M. Henning and S. Vinoski. *Advanced CORBA Programming with C++*. Addison-Wesley, 1999.
15. G. Brose, A. Vogel, and K. Duddy. *Java Programming with CORBA: Advanced Techniques for Building Distributed Applications*. OMG Press, 3rd edition, 2001.
16. J. L. Armstrong, M. C. Williams, C. Wikström, and S. R. Virding. *Concurrent Programming in Erlang*. Prentice Hall, 2nd edition edition, 1996.
17. Juan J. Sánchez Víctor M. Gulías, Carlos Abalde. Lambda goes to hollywood. In *Fifth International Symposium on Practical Aspects of Declarative Languages (PADL'03)*, volume 2562 of LNCS. Springer-Verlag, January 2003.
18. Thomas Arts and Clara Benac Earle. Verifying Erlang code: a resource locker case-study. In *Int. Symposium on Formal Methods Europe*, volume 2391 of LNCS, pages 183–202. Springer-Verlag, July 2002.
19. Thomas Arts and Juan José Sánchez Penas. Global scheduler properties derived from local restrictions. In *Proceedings of ACM Sigplan Erlang Workshop*. ACM, October 2002.

20. Ulf Wiger. Four-fold increase in productivity and quality; industrial-strength functional programming in telecom-class products. In *Workshop on Formal Design of Safety Critical Embedded Systems*, 2001.
21. Kathryn Fisler. *A Unified Approach to Hardware Verification Through a Heterogeneous Logic of Design Diagrams*. PhD thesis, Indiana University, 1996.
22. K. Havelund and N. Shankar. Experiments in theorem proving and model checking for protocol verification. In *Third International Symposium of Formal Methods*, volume 1051 of *LNCS*, pages 662–681. Springer-Verlag, 1996.
23. O. Marttila and P. Vuorimaa. XML based mobile services. In *Proceedings of 8 th Intl. Conf. in Central Europe Computer Graphics, Visualization, and Interactive Digital Media*, 2000.
24. A. Alexander. In *A Pattern Language: Towns, Buildings, Constructions*. Oxford University Press, 1977.

Partial-Order Reduction in Model Checking Object-Oriented Petri Nets

Milan Česka, Luděk Haša, and Tomáš Vojnar

Faculty of Information Technology,
Brno University of Technology
Božetěchova 2, CZ-612 66 Brno,
Czech Republic
{ceska,hasalud,vojnar}@fit.vutbr.cz

Abstract. The main problem being faced in finite-state model checking is the state space explosion problem. For coping with it, many advanced methods for reducing state spaces have been proposed. One of the most successful methods (especially when dealing with software systems) is the so-called partial-order reduction. In the paper, we examine how this method can be used in the context of object-oriented Petri nets, which bring in features like dynamic instantiation, late binding, garbage collection, etc.

1 Introduction

Model checking is an automatic technique for validating correctness of concurrent systems. Compared to the more traditional approaches based on simulation and testing, it allows a desired behavioral property to be checked over a model of the given system in such a way that all the reachable states that may affect the property may be guaranteed to be covered.

For building models of systems to be verified, we need some modelling formalism. Here, we consider the systems to be modelled by *object-oriented Petri nets* (OOPNs). We deal with the OOPNs associated with the PNTalk language and tool that have been developed at the Brno University of Technology [8]. PNTalk supports intuitive modelling of all key features of concurrent and distributed object-oriented systems, such as object-orientation, message sending, parallelism and synchronization. This is achieved by means of dealing with active objects encapsulating sets of processes described by high-level Petri nets, which communicate both via shared memory as well as message sending.

A big challenge in finite-state model checking is the *state space explosion problem*, i.e. the exponential growth of state spaces. Fortunately, there have been proposed many advanced methods for reducing state spaces. One of the most successful methods (especially when dealing with software systems) is the so-called *partial-order reduction*. In this paper, we examine how this method can be used in the context of OOPNs, which bring in features like dynamic instantiation, late binding, garbage collection, etc. We discuss how an application of partial-order reduction over OOPNs is complicated by these features. The algorithm we

then propose represents the first step towards exploiting partial-order reduction in on-the-fly model checking over OOPNs. It is relatively simple, but still quite useful in many cases. Moreover, it forms a basis for further improvements both from the point of view of finer reduction as well as coping with more complex verification tasks (more complex specification predicates, logics, etc.). In the paper, we consider verification of deadlockability and state invariants, and also verification against a class of global (not instance-oriented) LTL (linear-time temporal logic) formulae.

Related work. Partial-order reduction methods were developed independently by several researchers. They can be found with different names assigned to the set of transitions that cannot be postponed: *ample sets* [2], *persistent sets* [5], and *stubborn sets* [11]. Partial-order reduction over different dialects of high-level Petri nets, such as Coloured Petri Nets (CPNs) [9], and over models with a dynamic creation of instances, as in Java PathFinder [1] and Spin [6], have been particularly inspiring for our work, because features of such systems are combined in OOPNs. Our previous works on generating and using state spaces of OOPNs (e.g., [4,12]) include symmetry-based reduction techniques for coping with state space redundancies related to dealing with identifiers of dynamically arising and disappearing net instances, methods for querying over state spaces of OOPNs where the structure of states is not known in advance, methods for effective garbage collection and calculation of enabled events, etc.

Outline. The rest of the paper is organized as follows. Section 2 informally introduces OOPNs. Section 3 provides a general view on the partial-order reduction method inspired by [2]. Partial-order reduction over OOPNs is discussed in Section 4. Section 5 concludes the paper and overviews the future work.

2 Key Concepts of OOPNs

The OOPN formalism is characterized by a Smalltalk-based object-orientation enriched with concurrency and polymorphic transition execution, which allows message sending, waiting for and accepting responses, creating new objects, and performing computations.

In the following, we explain the main principles of the structure and behaviour of OOPNs. A deeper introduction to the OOPN formalism can be found in [3] and the formal definition of OOPNs in [12].

2.1 The Structure of OOPNs

An *object-oriented Petri net* is defined on a collection of elements comprising constants, variables, net elements (i.e. places and transitions), class elements (i.e. object nets, method nets, synchronous ports, and message selectors), classes, object identifiers, and method net instance identifiers. An OOPN has its initial

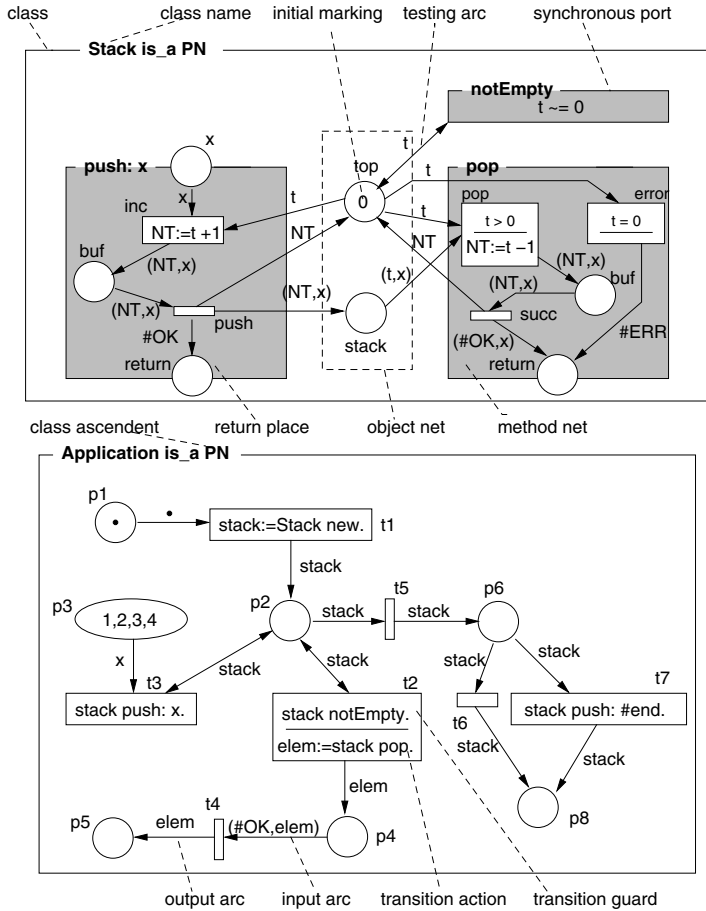


Fig. 1. An example of an OOPN model representing a simple system with a stack

class and initial object identifier, as well. The so-called universe of an OOPN contains (nested) tuples of constants, classes, and object identifiers. An OOPN class is given by its object net, its sets of method nets and synchronous ports, and a set of message selectors corresponding to its methods and ports.

Object nets consist of places and transitions. Each place has some (possibly empty) initial marking. Each transition has conditions and preconditions (i.e. inscribed testing and input arcs), a guard, an action, and postconditions (i.e. inscribed output arcs). Object nets describe what data particular objects encapsulate and what activities the objects may exhibit on their own.

Method nets resemble object nets but, additionally, each of them has a set of parameter places and a return place. Method nets can access places of the appropriate object nets, which allows running methods to modify the states of the objects which they are running in. Method nets specify how objects asynchronously respond to received messages.

Synchronous ports are special transitions that cannot fire alone but only dynamically fused to some regular transitions. These transitions (possibly indirectly) activate the ports via message sending from their guards. Each port has a set of conditions, preconditions, and postconditions over places of the appropriate object net, a guard, and a set of parameters. Parameters of an activated port s can be bound to constants or unified with variables from the level of the transition or port that activated s . Synchronous ports allow us to remotely test and change states of objects in an atomic way.

Fig. 1 shows an example of an OOPN-based model. Class **Stack** represents a stack with an unbounded buffer and class **Application** a model of a system using a stack. Class **Stack** has two methods, **push** and **pop**, and synchronous port **notEmpty**. Method **push** puts a token from its parameter place into the stack. It works in two steps. Firstly, the pointer to the top of the stack is removed from place **top** and incremented. Secondly, the pointer is returned to **top** and the tuple with the element and its position in the stack is stored in place **stack**. Method **pop** removes and returns the top element from the stack and also works in two steps. The synchronous port tests emptiness of the stack.

2.2 The Dynamic Behaviour of OOPNs

A state of an OOPN can be encoded as a marking, which can be structured into a system of objects. Thus the dynamic behaviour of OOPNs corresponds to an evolution of a system of objects. An object of a class c is a system of net instances that contains one instance of the object net of c and a set of currently running instances of method nets of c . Each net instance entails its identifier and a marking of its places and transitions. A marking of a place is a multiset of tokens coloured by some elements of the universe. A marking of a transition t is a set of records about methods invoked from t and not yet terminated.

For a given OOPN, its initial marking represents a single, initially marked object net instance from the initial class. A change of a marking of an OOPN is the result of occurrence of some event. Such an OOPN event $E = (e, id, t, b)$ is given by (1) its type e , (2) the identifier of the net instance id , it takes place in (3) the transition t it is statically represented by, and (4) the binding tree b containing the bindings of the variables used on the level of the involved transition as well as within all the synchronous ports (possibly indirectly) activated from that transition. There are four kinds of events according to the way of evaluating the action of the appropriate transition: **A**—an atomic action involving trivial computations only, **N**—a new object instantiation via the message **new**, **F**—an instantiation of a Petri-net described method, and **J**—a method net instance termination. Firing an **A** event means removing and/or adding some tokens from/to the marking of certain places according to the arcs of the involved transition and synchronous ports and according to the appropriate binding. An **N** event differs from an **A** event by additionally creating and initializing a new object. An **F** event starts a new method, initializes it, and puts arguments into its parameter places. The firing transition is marked by a reference to the new method net instance and its output arcs are ignored. A **J** event retrieves a result token from

the return place of a method net instance, deletes the instance and the transition marking element referencing it, and performs the output part of the appropriate transition. Garbage collection is a part of every event.

3 Partial-Order Reduction

The *partial-order reduction* method is aimed at reducing the size of the state space that needs to be searched by model checking algorithms. A state space is a structure consisting of the states that a system can reach and of transitions that can be done between these states in the system. The model checking algorithm is applied during generation of the reduced state space. The behaviours present in the reduced state space are a subset of the behaviours in the full state space. A justification of the reduction method shows that the behaviours that are not present do not affect the property being checked.

The partial-order reduction method exploits commutativity of concurrently executable transitions whose firing results in the same state regardless of the firing order. When a specification cannot distinguish between two interleaving sequences that differ only in the order in which concurrently executable transitions are taken, it is sufficient to analyze only one of them. Thus, we always try to fire only a subset of the enabled transitions. When it is too hard to check whether firing some transition may be postponed or omitted or not, assuming that it must be fired preserves correctness of the reduction.

The partial-order reduction technique is best suited for asynchronous concurrent systems where most activities taken by different processes are performed independently. OOPNs fall into the class of such systems.

3.1 Labeled Transition Systems

State spaces of OOPNs can be viewed as labeled transition systems. A finite *labeled transition system* (LTS) is a tuple (S, T, S_0, L, AP) where S is a finite set of states, S_0 is the set of initial states, T is a finite set of transitions such that for each transition $\alpha \in T$, $\alpha \subseteq S \times S$. AP is a finite set of atomic propositions, and L is a labeling function $S \rightarrow 2^{AP}$. For a given OOPN, LTS states correspond to reachable markings and LTS transitions to applicable events.

An LTS transition $\alpha \in T$ is enabled in a state s if there is a state s' such that $\alpha(s, s')$ holds. Let the set of transitions enabled in s be $enabled(s)$. A transition α is deterministic if for every state s , there is at most one state s' such that $\alpha(s, s')$, and then we use $s' = \alpha(s)$ instead of $\alpha(s, s')$.

For a given LTS, a path from a state s is defined as a finite or infinite sequence of states interleaved with transitions, i.e. a sequence $s_0, \alpha_0, s_1, \dots$ such that $s = s_0$ and for every i , $s_{i+1} = \alpha_i(s_i)$.

Algorithms for partial-order reduction are often based upon modifying the depth-first search. For each state s , they select and follow only a subset of the enabled transitions $enabled(s)$ called an *ample set* and denoted by $ample(s)$.

The model checking algorithm is applied either to the resulting LTS or on-the-fly during generation. Here, we adopt the more common second approach.

Partial-order techniques are based on the observation that the order in which some transitions are executed is not relevant. This leads to the concept of independence between transitions that can be executed concurrently. An *independence* relation $I \subseteq T \times T$ is a symmetric, antireflexive relation, satisfying the following two conditions for each state $s \in S$ and for each $(\alpha, \beta) \in I$: *enabledness*—if $\alpha, \beta \in \text{enabled}(s)$, then $\alpha \in \text{enabled}(\beta(s))$, and *commutativity*—if $\alpha, \beta \in \text{enabled}(s)$, then $\alpha(\beta(s)) = \beta(\alpha(s))$. The *dependency* relation D is the complement of I , $D = (T \times T) \setminus I$.

Another concept that plays a significant role in the reduction is *invisibility*. A transition $\alpha \in T$ is invisible wrt. a set of propositions $AP' \subseteq AP$ if for each pair of states $s, s' \in S$ such that $s' = \alpha(s)$, $L(s) \cap AP' = L(s') \cap AP'$. In other words, a transition is invisible if its firing does not change the value of the propositional variables in AP' .

Finally, let us present the concept of *stuttering equivalence*. A *block* is defined as a finite sequence of identically labeled states. Two paths are stuttering equivalent if they can be defined as a concatenation of blocks, and, for each $k > 0$, the k -th block in one of the paths has the same label as the k -th block in the other. Two systems are stuttering equivalent if and only if they have the same set of initial states and for each path in one of the systems starting from an initial state, there exists a stuttering equivalent path in the other system starting from the same initial state.

3.2 Linear Temporal Logic

Temporal logics differ from classical logics in having temporal operators allowing one to refer to temporal relations of states and transitions. Linear temporal logic (LTL) is a temporal logic whose formula holds iff it holds for every infinite execution of a system (deadlocked executions may be extended to infinite ones by repeating their terminal state).

Given a finite, non-empty set of atomic propositions AP , LTL formulae are defined inductively as follows: (1) Every $p \in AP$ is an LTL formula. (2) If φ and ψ are LTL formulae, then so are $\neg\varphi$, $\varphi \vee \psi$, $\bigcirc\varphi$, and $\varphi \mathcal{U}\psi$. (3) There are no other LTL formulae.

Let $\langle P_0, P_1, \dots \rangle$ be an infinite sequence of subsets of atomic propositions and let φ be a linear temporal logic formula, then:

- $\langle P_0, P_1, \dots \rangle \models \varphi$ iff $\varphi \in P_0$. Moreover, propositional operators could be introduced in the usual way here.
- $\langle P_0, P_1, \dots \rangle \models \bigcirc\varphi$ iff $\langle P_1, P_2, \dots \rangle \models \varphi$.
- $\langle P_0, P_1, \dots \rangle \models \varphi \mathcal{U}\psi$ iff there exists $i \geq 0$ such that $\langle P_i, P_{i+1}, \dots \rangle \models \psi$ and $\langle P_j, P_{j+1}, \dots \rangle \models \varphi$ for each j such that $0 \leq j < i$.

In this paper, the sublogic of LTL called *next-time-free* LTL ($\text{LTL} \setminus X$) excluding the $\bigcirc$ operator is used. The common abbreviations $\Diamond \equiv \text{true } \mathcal{U} \varphi$ (eventually/sometimes) and $\Box\varphi \equiv \neg\Diamond\neg\varphi$ (always/henceforth) are allowed too. It can

be shown that $LTL \setminus X$ formulae cannot distinguish between stuttering equivalent transitions systems [2].

3.3 Ample Set Conditions for $LTL \setminus X$

To ensure correctness of $LTL \setminus X$ verification, ample sets must be constructed in such a way that the reduced state space is stuttering equivalent to the full state space. There are four conditions for selecting $ample(s) \subseteq enabled(s)$ such that this goal is achieved [2]. The reduction depends on the set of propositions AP' that appear in the $LTL \setminus X$ formula.

- **Condition C0:** $ample(s) = \emptyset$ iff $enabled(s) = \emptyset$.
- **Condition C1:** Along every path in the full LTS that starts at s , the following condition holds: A transition that depends on a transition in $ample(s)$ cannot be executed without a transition in $ample(s)$ occurring first.
- **Condition C2:** If s is not fully expanded, every $\alpha \in ample(s)$ is invisible.
- **Condition C3:** A cycle is prohibited if it contains a state in which some transition is enabled but is never included in $ample(s)$ for any s on the cycle.

Conditions **C0**, **C2** are easy to check and do not depend on the search algorithm. Condition **C1** is also independent of the search algorithm, but it is the most complicated one among the constraints. Condition **C3** depends on the search algorithm. For the depth-first search (DFS), we strengthen **C3** as follows:

- **Condition C3':** If s is not fully expanded, then no transition in $ample(s)$ may reach a state that is on the DFS stack.

3.4 On-the-Fly Reduction

The partial-order reduction can be used in conjunction with on-the-fly model checking. It is more efficient to combine the construction of a state space with checking the correctness of a specification. It is often possible to identify that the system violates the specification before completing the construction of the state space, and even if not, some parts of the state spaces may be detected not interesting for the validity of the specification.

In this paper, we consider verification of deadlocks, state invariants, and properties described by means of $LTL \setminus X$. The model checking problem for OOPNs and $LTL \setminus X$ can be solved by viewing the state space of an OOPN (represented as an LTS) as a Büchi automaton with all states accepting, by translating a given specification to a Büchi automaton, and by checking that the language of the product of the system and the specification is empty. The emptiness problem may be solved via a nested DFS [7]. For verification of deadlocks and invariants, we can simplify this approach as discussed later on.

4 Partial-Order Reduction in Model Checking OOPNs

4.1 Querying State Space in OOPNs

For describing invariants and atoms of temporal logic formulae specifying properties of OOPN-based models to be checked, we need a state space query language over state spaces of OOPNs that is able to cope with the dynamically changing structure of states of OOPNs. Indeed, sets of the existing instances and their mutual relations can be different in every encountered state and cannot be fully predicted.

Two groups of functions may be used as a basis of an OOPN state space query language [12]. The first group includes the so-called *instance querying functions*. They allow us to obtain sets of the currently existing instances of certain nets. Next, they make it possible to recursively derive sets of the net instances or constants that are directly or transitively referred from some already known instances via the marking of some of their places and transitions. The second group of functions consists of the so-called *set iterating functions*. They allow us to examine and process the sets of instances and constants obtained from the instance querying functions. The functions from these two groups are intended to be combined in order to obtain the required characteristics of states.

In the form of Prolog predicates used in the prototype tool for exploring state spaces of OOPNs [12], we will use here the following instance querying functions that all take the current state to be implicit and return results via their last parameter: Predicate `init(Is)` returns the set with the initial object net instance. Predicate called `inst(Cs,Ns,Is)` returns the set of currently existing instances belonging to the nets from `Ns` and running within objects that belong to the classes from `Cs`. Predicate `token(Is,Ps,Cs,Ms)` returns the set of the tokens that belong to the classes from `Cs` and are stored in the places from `Ps` within the instances from `Is`. There is also a multiset version of `token` called `mtoken`. Predicate `invoc(Is,Ts,Cs,Ns,Bs)` returns the set of method net invocations that were launched by the transitions belonging to `Ts` from the instances from `Is`. Only the invocations are selected which correspond to instances of the nets from `Ns` running within objects of the classes from `Cs`. Predicate `within(Is1,Ns,Is2)` collects all the instances of the nets from `Ns` which run within the objects from `Is1`. We except here predicate `ref` defined in [12] and making accessible instances transitively referenced via any anonymous instances. The reason is that the definition of invisibility for this predicate is problematic and so-far open.

From the group of set iterating functions, predicate `sforall(S,X,P,Y)` returns `true` iff predicate `P` over `X` is fulfilled for all elements of non-empty set `S`. The weak version called `wforall` is fulfilled over the empty set. Predicate `size(S,X)` (and its multiset variant `msize`) returns the number of elements in (multi)set `S`. Predicate `select(S1,X,P,S2)` selects all the elements `X` from `S1` that fulfill predicate `P` over `X`.

Using the described query functions, we can, e.g., define an invariant requiring place `top` from Fig. 1 to always contain nothing or a single token referring to the top of the stack:

```

Top :- inst([stack],[[stack,object]],I),
      wforall(I,X,(mtoken([X],[stack],all,MS1),msize(MS1,N),
                    (mtoken([X],[top],all,[]);
                     mtoken([X],[top],all,[1,N])))),true).

```

As another example, we can define predicate `StackIsEmpty`

```

StackIsEmpty :- inst([Stack],[[Stack,object]],I),
                mtoken(I,[stack],all,[]).

```

and pose an LTL $\setminus X$ query $\Box \Diamond \text{StackIsEmpty}$ (which does not hold in our case).

4.2 Independence and Invisibility in OOPNs

The main problem in computing ample sets is handling of **Condition C1**. It is, of course, not feasible to first generate the entire state space and then choose ample sets satisfying **C1**. Ample sets must be computed on-the-fly by carefully selecting LTS transitions (OOPN events) to be fired such that they are independent of the omitted, currently enabled transitions and also such that it is guaranteed by the static structure of the given OOPN and by the current state (without firing some further transitions) that no transition can become fireable without any transition from the ample set having been fired.

The dependency analysis needed when computing ample sets is probably always difficult. However, it seems to be particularly difficult in OOPNs where problems with partial order reduction arising in high-level Petri nets and in dynamically structured formalisms are combined.

As usual in high-level Petri nets, a token from a particular place can occur in bindings of multiple events out of which some may be enabled at a given state and the others may become enabled only after some steps of the OOPN evolution. In many cases, it is hard to exactly know all the kinds of tokens that can get into the input places of transitions and ports. Thus, it is hard to determine all the possible bindings, and so all the dependent events.

A specialty of OOPNs is synchronous ports that can be called from guards. Due to the dynamism and late binding in OOPNs, it is hard to know which instances will be able to call which synchronous ports and when, and thus which transitions may have bindings that yield events not colliding via their transitions, but via some synchronous ports. In the algorithm presented here as the initial approach to partial-order reduction in OOPNs, we exclude events calling synchronous ports from ample sets not containing all enabled events. Another problem arises when an event destroys some net instance. Then, the garbage collector could rule out some postponed events. Therefore, we currently do not allow such events in non-trivial ample sets. Next, we require all input places and the transition of an event from a non-trivial ample set to lie in a single net. Otherwise, dynamic creation of new net instances could additionally yield new colliding events based on the same transition, but within different instances. For the future, the problems listed here represent some of the points where our algorithm could be refined.

The notion of invisibility in OOPNs is also quite complex in general. For example, in predicate **ref** proposed as a part of the query language in [12], we may have to deal with indirect references between visible instances going through places and nets not at all specified in advance. However, for the predicates we chose above, the situation is easier. Both for checking invariants and $LTL \setminus X$ formulae based on them, we may define an event to be invisible if it does not change the marking of any place/transition that occurs in the specification of the property being checked and does not create nor remove any net instance that occurs in the specification.

More precisely, an event is invisible if it does not change the marking of any place occurring within a predicate **mtoken** or **token** used in a given query. The event is invisible if it does not create nor remove a net instance that occurs within a predicate **inst** or **within**. For predicate **invoc**, the events are visible that belong to the transitions used in the predicate and create or remove an instance of a net mentioned in the predicate. This notion of invisibility is, of course, not accurate. An event is sometimes considered visible though it is not necessary. A finer notion of invisibility is an interesting subject for the future research.

4.3 Computing Ample Sets in OOPNs

For every event e , we define its *loss set*, $loss(e)$, as the set of the variables that appear in the inscriptions of the involved transitions, ports, and the adjoint arcs, and about whose contents we cannot easily say that it will not be lost when firing the event. The formal definition can be found in [12]. Here, it suffices to note that for an **A** or **N** event not calling any port, the loss set includes the variables of the involved transition that appear on its input arcs (or as a result variable in its action), but not on its output or testing arcs. For **F** events not calling any port, the loss is set empty—the contents of the transition variables is saved in the resulting transition marking element. (We will not need the notion of loss sets for **J** events here.) Next, let the notation $\bullet x$ denote the set of the input places, transitions, or synchronous ports of a place, transition, or synchronous port x . Let the notion of $x\bullet$ be introduced in a similar way.

We can now describe the algorithm we propose here for computing an ample set in a state s of an OOPN as follows:

-
1. Try to find an enabled event $E = (e, id, t, b)$ such that $e \neq J$, $\bullet t = \{p\}$, $p\bullet = \{t\}$, t has either an input or a testing arc only, and a single token is being removed/tested by it. If successful, let $ample = \{E\}$. If unsuccessful, go to 2, else go to 5.
 2. Try to find an enabled event $E = (e, id, t, b)$ with $e \neq J$ such that the expression of each input/testing arc of t does not involve any variable. Create an auxiliary set $\mathcal{T}$ and add t into $\mathcal{T}$. Go on by (a).
 - a) For a transition $r \in \mathcal{T}$, add all the transitions $u \in (\bullet r)\bullet$ into $\mathcal{T}$ for which there is no place $p \in \bullet u$ with the empty marking and $p\bullet = \emptyset$.

- b) Repeat step (a) for transitions that are newly added into $\mathcal{T}$ until no new transition is added.

If the expression of each input/testing arc of transitions in $\mathcal{T}$ does not involve any variable and for each transition in $\mathcal{T}$, there is an enabled, non-J event in the net instance id , add all these enabled events to the ample set and go to step 5, else repeat step 2 until no new event can be chosen. Go to 3.

3. Try to find an enabled event $E = (e, id, t, b)$, $e \neq J$, such that each input/testing arc of t may remove/test a single token only and this token must always be the same for all the arcs. Create an auxiliary set $\mathcal{T}$ and add t into $\mathcal{T}$. Go on by (a).
 - a) For a transition $r \in \mathcal{T}$, add all the transition $u \in (\bullet r)\bullet$ into $\mathcal{T}$ for which there is no place $p \in \bullet u$ with the empty marking and $\bullet p = \emptyset$.
 - b) Repeat step (a) for transitions that are newly added into $\mathcal{T}$ until no new transition is added.

If each transition in $\mathcal{T}$ also fulfills the above condition for t , and for each transition in $\mathcal{T}$, there is an enabled, non-J event in the net instance id such that the token being removed/tested is the same for all the events, add all these enabled events to the ample set and go to step 5, else repeat step 3 until no new event can be chosen. Go to 4.

4. Try to find an enabled, non-J event $E = (e, id, t, b)$ such that $(\bullet t)\bullet = \{t\}$ and t has two types of arcs only: (1) An input/testing arc of t either removes/tests a single token and the preset of the input place of this arc is empty, or (2) the arc expression of the arc does not involve any variable. Add all enabled, non-J events of t to the ample set. If no event is taken, return the set of all enabled events, else go to step 5.
5. If $ample(s) \subset enabled(s)$, then for all events in the ample set, check **Condition C2** and **C3'** and whether their loss sets are empty, their guards do not involve calling synchronous ports, they do not remove any token from any return place, and their transitions and input places lie in a single net. If these conditions are violated, try to find a new ample set—return to the step that was done the last time and start it with another event E . Otherwise, return the ample set.

The algorithm for computing $ample(s)$ tries to find a small set of enabled events that satisfies all conditions for ample sets. As discussed already in Section 4.2, in order to simplify achievement of the guarantee that **C1** is not broken, we always consider only the events that do not call synchronous ports from guards, do not destroy any object or method net instance (which is always the case of J events, which we therefore exclude from non-trivial ample sets), and do not involve transitions linked with places from a different net. Due to the difficulties of handling J events destroying method net instances (and may be also some other instances referenced from them only), we as well exclude any interference of events of non-trivial ample sets with J events.

In Step 1 of the algorithm, we try to select an event E based on a transition t that has only one input/testing arc removing/testing a single token from an input place p . Moreover, p has only t in its postset. Thus, there is no other event which could use the token in p being removed/tested by E . (Note that allowing an arc with a variable number of tokens to be consumed could yield conflicting events. On the other hand, the condition could be generalized to removing/testing any constant number of tokens.)

In Step 2, an event based on a transition t whose input/testing arc expressions are constant is selected. We add t to an auxiliary set $\mathcal{T}$. Then, iteratively, all conflicting transitions of the transitions in $\mathcal{T}$ are added to $\mathcal{T}$. The transitions that have an input place with the empty marking (which could be generalized to a marking not containing the concerned token) and with the empty preset are not added because there is no more an enabled event based on them that could interfere with the other considered events. If each transition from $\mathcal{T}$ has input/testing arcs with exclusively constant expressions, and an enabled event over it exists in the concerned instance, these enabled events form an ample set—there is no other event outside the ample set that could collide with the selected ones.

Step 3 is similar to Step 2, but all input/testing arcs of the conflicting transitions must remove/test a single token that is statically guaranteed to be the same for all the arcs of a single transition due to using the same constant, variable, or tuple. In the ample set, we cover all events over these transitions that could collide when choosing the same token value to be handled.

In Step 4, we consider a transition with input places that have output to this transition only. They either remove/test a constant token or are linked to a place without any input. Since new tokens can appear only in the input places from which a fixed type of tokens is taken, no new events can collide with the currently enabled ones in the future.

4.4 On-the-Fly Model Checking with Reduction

In the previous section, the ample set construction for generating reduced OOPN state space was explained in a way parameterized by the notion of invisibility. We now specialize it for the different properties to be verified we consider here and combine it with the appropriate state space generating and verification procedures.

Deadlocks. For verification of deadlocks, a simple (not nested) DFS state space exploration algorithm may be used. Instead of always firing all enabled transitions, only the ones in the ample sets computed according to Section 4.3 are fired. During the generation of a state space, in every new state, we check whether the set $enabled(s)$ is not empty—if it is empty, a deadlock is announced, the verification is stopped, and a counterexample trace is taken from the DFS stack.

If $ample(s) \subset enabled(s)$, the definition of ample sets implies that there is an event $E \in enabled(s) \setminus ample(s)$ that is enabled in the next states even

when **Condition C2** is ignored. (Otherwise, E would depend on some events in $\text{ample}(s)$, and **C1** would be broken.) Thus, we do not need to check **C2** when computing the ample sets.

Invariants. For verification of invariants, a simple DFS algorithm with firing only the transitions from the ample sets computed according to Section 4.3 may again be used. The notion of invisibility defined in Section 4.2 is used when checking **Condition C2**. The invariant is being evaluated in every reached state. If it is violated, an error is announced, and the counterexample trace is output.

LTL\X. We can combine a use of the ample sets computed according to Section 4.3 with the nested DFS algorithm [7]. The notion of invisibility defined in Section 4.2 is used when checking **Condition C2** (taking into account all the OOPN state space query predicates appearing in the given formula).

5 Conclusion

We have identified and discussed problems that are to be faced when applying the partial-order reduction technique in model checking over OOPNs associated with the PNTalk language and tool. It turns out that the problems that arise here are quite complex because OOPNs combine various obstacles to the application of the partial-order reduction technique that show up in the domain of high-level Petri nets and in the domain of systems with dynamic instantiation. The problems are due to it is hard to estimate which bindings of which transitions may occur and when (and thus which transition bindings should be taken into account in a dependency analysis), which transitions may call which synchronous ports and when (causing different indirect conflicts), which transition instances sharing a certain place may arise and when, which instances may be deleted and when, etc.

We have proposed an algorithm for partial-order reduction in model checking deadlockability and certain useful classes of invariants and global LTL\X formulae over OOPNs (given by the allowed OOPN state query predicates). The algorithm represents the first attempt to apply partial-order reduction in the context of state spaces of OOPNs. It is relatively simple, yet we believe it can yield a significant reduction in many cases (e.g., multiple non-trivial ample sets may be generated when verifying the stack example from Fig. 1).

The proposed algorithm can be used as a basis for further improvements both to achieve a better reduction and to allow for more complex verification tasks. Suitable directions for improving the reduction are, e.g., trying to allow at least some events calling synchronous ports, events destroying some instances, or events with transitions with input places distributed in several nets to appear in non-trivial ample sets. As for more complex verification tasks, it remains to cover some more sophisticated OOPN state query predicates (such as **ref** mentioned in Section 4.1) and to consider verification of properties local to particular instances

(expressible via some indexed temporal logic allowing particular instances to be tracked across state spaces).

The here proposed algorithm is currently being implemented in a prototype way in a tool for verification of OOPNs, and we hope the experience gained from experimenting with it will also help us in the further refinements of the methods proposed here.

Acknowledgment. This work has been supported by the Czech Ministry of Education within the project FR829/2003/G1 “State Space Reductions for Object-Oriented Petri Nets” and is a part of the research project No. CEZ:J22/98:262200012 “Research in Information and Control Systems”. It was also supported by the Czech Grant Agency under the contract 102/01/1485 “Environment for Development, Modelling, and Application of Heterogeneous Systems”.

References

1. G. Brat, K. Havelund, S. Park, and W. Visser. Java PathFinder – A Second Generation of a Java Model Checker. In *Workshop on Advances in Verification*, 2000.
2. E.M. Clarke, O. Grumberg, and D.A. Peled. *Model Checking*. The MIT Press, Cambridge, Massachusetts, London, England, 2000.
3. M. Češka, V. Janoušek, and T. Vojnar. PNtalk – A Computerized Tool for Object-Oriented Petri Nets Modelling. In *Proc. EUROCAST’97*, vol. 1333 of *LNCS*, 1997. Springer-Verlag.
4. M. Češka, V. Janoušek, and T. Vojnar. Generating and Using State Spaces of Object-Oriented Petri Nets. *International Journal of Computer Systems Science and Engineering*, 16(3):183–193, 2001.
5. P. Godefroid. *Partial-Order Methods for the Verification of Concurrent Systems – An Approach to the State-Explosion Problem*. PhD thesis, University of Liege, Computer Science Department, 1994.
6. G. J. Holzmann, and D. Peled. An Improvement in Formal Verification. In *Proc. of 7th FORTE International Conference on Formal Description Techniques, IFIP Conference Proceedings*, 1994. Chapman & Hall.
7. G.J. Holzmann, D. Peled, and M. Yannakakis. On Nested Depth First Search. In *Proc. of the 2nd Spin Workshop*, American Mathematical Society, 1996.
8. V. Janoušek. *Modelling Objects by Petri Nets*. PhD. thesis, Brno University of Technology, Brno, CZ, 1998.
9. L.M. Kristensen and A. Valmari. Finding Stubborn Sets of Coloured Petri Nets Without Unfolding. In *ICATPN’98*, vol. 1420 of *LNCS*, 1998. Springer-Verlag.
10. A. Lluch-Lafuente, L. Edelkamp, and S. Leue. *Partial Order Reduction in Directed Model Checking*. Technical report, 2001.
11. A. Valmari. The State Explosion Problem. In *Lectures on Petri Nets I: Basic Models*, vol. 1491 of *LNCS*. Springer-Verlag, 1998.
12. T. Vojnar. *Towards Formal Analysis and Verification over State Spaces of Object-Oriented Petri Nets*. PhD. thesis, Brno University of Technology, Brno, CZ, 2001.

On the Strong Co-induction in Coq^{*}

J.L. Freire Nistal¹, A. Blanco Ferro¹, Victor M. Gulías¹, and E. Freire Brañas²

¹ LFCIA, Campus de Elviña
15071, La Coruña, Spain
{freire, blanco, gulas}@lfcia.org
<http://www.lfcia.org>
² IES 1, Oleiros
15173 Oleiros, Spain
rike@inicia.es

Abstract. In this paper, we provide a library in Coq containing intuitionistic proofs of some facts that are on the basis of formal verification tools such as Model Checking or Theorem Proving: the Reduction Lemma [8] [17] and the correspondent on minimum fixed points [1]. In order to improve usability, most of the proofs are given in a general frame of partial order relations and not only in the specific complete lattice of a power-set.

1 Introduction

The theorems of fixed point are very useful in computer science providing semantics of programming languages, program analysis, program verification, etc.

To say a set is inductively defined just means it is the least fixed point of some monotone function. For instance, for some universal U , suppose there is an element $O \in U$ and an injective function $S : U \rightarrow U$. If we define a monotone function $F : 2^U \rightarrow 2^U$ by

$$F(X) = \{O\} \cup \{S(x) \mid x \in X\}$$

then $\mathbb{N} = \mu X.F(X)$ defines the set of naturals. The associated principle of induction is that $\mathbb{N} \subseteq X$ if $F(X) \subseteq X$, which is to say that $\mathbb{N} \subseteq X$ if both $O \in X$ and $S(x) \in X$ whenever $x \in X$.

Also to show equivalence of functional programs which possibly consume and generate infinite data-type structures, applicative bisimulation is defined co-inductively [6] [2]. The principles of co-induction and strong co-induction provide a method to find some monotonic function $\phi^\equiv$ capturing the meaning of the equivalence and define $\equiv$ to be the greatest fixed point of that function $\nu X.\phi^\equiv(X)$. To prove that $x \equiv y$ by co-induction, we only need to find some relation R such that $x R y$ and prove that $R \subseteq \phi^\equiv(R)$.

^{*} Supported by projects: Xunta Galicia: PGIDIT02TIC00101CT and Spanish Gov. MCyT:TIC 2002-02859.

This paper is part of a large project which has as goal to model a concurrent system and its logical properties in **Coq**. This includes the formalization of the modal μ -calculus and also a model of the concurrent system itself. Here, we provide a library in **Coq** containing intuitionistic proofs of some facts that are on the basis of formal verification tools such as Model Checking and Theorem Proving: the principle of co-induction, the principle of strong co-induction or Reduction Lemma [8] [17] and the Well-founded induction on minimum fixed points [1].

In order to improve usability, most of the proofs are given for partial order relations in general and not only for the specific power-set complete lattice.

2 The Coq System: An Overview

The logical framework **Coq** is an implementation of the Calculus of Inductive Constructions (CIC) of G. Huet, T. Coquand and C. Paulin-Mohring. Developed at INRIA, it is a goal-directed and tactic-driven theorem prover where types can be defined inductively. It has a set of predefined tactics, including an **Auto** tactic which tries to apply previous proofs. The default logic is intuitionistic but classical logic is also available by requiring the **Classical** module.

The system automatically extracts the constructive contents of proofs as an executable ML program that permits the development of programs consistent with their specification.

The notation **a**:**A** (**a** is of type **A**) is interpreted as "**a** is a proof of **A**", when **A** is of type **Prop**, or "**a** is an element of the specification **A**", when **A** is of type **Set**. Here, **Prop** and **Set** are the impredicative types of the system. These two types and a hierarchy of universes **Type**(*i*) for any natural *i*, are the elements of the set of sorts. The sorts have the following properties: **Prop**:**Type**(0) and **Type**(*i*):**Type**(*i* + 1).

Allowed constructions are: **x** | (**M N**) | [**x**:**T**] **f** | (**x**:**T**)**U**, where **x** denotes variables as well as constants; (**M N**) is the application; the third expression represents the program (λ -expression) of parameter **x** and body the term **f** (the abstraction of variable **x** of type **T** in **f**). Lastly, the fourth is the program type that admits an entry of type **T** and returns a result of type **U**. This type is referred to as *product type* and, in type theory, is represented as $\prod x : T. U$ or also as $\forall x : T. U$. If **x** is not free in **U**, then this is simply written $T \rightarrow U$, the type of the functions between these two types, or non-dependent product.

Typing rules provide also proof tactics when reading bottom up. For example:

$$\frac{E[\Gamma] \vdash (x : T)U : s \quad E[\Gamma :: (x : T)] \vdash f : U}{E[\Gamma] \vdash [x : T]f : (x : T)U} \text{Lam}$$

expresses that the term (program) $[x : T]f$ has the product type $(x : T)U$ provided that this type has type sort, and if in the environment *E* and context *Γ* with the additional hypothesis $x : T$, the term *f* has type *U*.

If we start with the type $(x : T)U$ and look for some term which inhabits it, we can use the *Intro* tactic to obtain the subgoal *U*. If we can construct *f* of

type U then Coq itself builds the term $[x : T]f$ which, because of the *lam* rule, will have type $(x : T)U$.

Inductive Types. Under certain constraints inductive types can be defined in the system Coq and with each of them a structural induction principle, and possibly a recursion scheme, are automatically generated by the system. For example the type of natural numbers $\mathbb{N}$:

```
Coq < Inductive nat:Set := 0:nat | S:nat -> nat.
nat is defined
nat_ind is defined
nat_rec is defined
...

Coq < Parameters P:nat->Set;o:(P 0);h:(n:nat)(P n)->(P (S n));n:nat.

Coq < Eval Compute in (nat_rec P o h 0).
      = o
      : (P 0)

Coq < Eval Compute in (nat_rec P o h (S n)).
      = (h n (nat_rec P o h n))
      : (P (S n))
```

2.1 Sets and Orders in Coq

In the module Ensembles of the Coq distribution library, the basic facts of Set theory on the type `Type` are implemented. Given some universal $U : \text{Type}$, a set A is defined as a predicate on U .

```
Coq < Require Ensembles.

Coq < Print Ensemble.
Ensemble = [U:Type]U->Prop
           : Type->Type

Coq < Print In.
In =
[U:Type; A:(Ensemble U); x:U](A x)
   : (U:Type)(Ensemble U)->U->Prop
```

Therefore, if $A : (\text{Ensemble } U)$ and $x : U$, then $(\text{In } U \ A \ x) == (A \ x)$.

```
Coq < Print Included.
Included =
[U:Type; B,C:(Ensemble U)](x:U)(In U B x)->(In U C x)
   : (U:Type)(Ensemble U)->(Ensemble U)->Prop
```


Next, the couple set $\{x, y\}$ and the full set on U are defined:

```
Coq < Inductive Couple [U : Type; x : U; y : U] : (Ensemble U) :=
Coq <      Couple_l : (In U (Couple U x y) x)
Coq <      | Couple_r : (In U (Couple U x y) y).
Couple is defined
Couple_ind is defined

Coq < Inductive Full_set [U : Type] : (Ensemble U) :=
Coq <      Full_intro : (x:U)(In U (Full_set U) x).
Full_set is defined
Full_set_ind is defined
```

Relations and its properties, like for example, transitivity, are coded in the module `Relations_1`:

```
Relation = [U:Type]U->U->Prop
          : Type->Type

Transitive =
[U:Type; R:(Relation U)](x,y,z:U)(R x y)->(R y z)->(R x z)
  : (U:Type)(Relation U)->Prop

Inductive Order [U : Type; R : (Relation U)] : Prop :=
  Definition_of_order : (Reflexive U R)
                        ->(Transitive U R)
                        ->(Antisymmetric U R)
                        ->(Order U R)
```

Well-founded Relations. Well-founded relations are the essence of induction. In particular, they are crucial for establishing the absence of infinite loops in a program. Well-founded relations are also used for proving termination of rewriting systems.

The predicate `acc` implements in Coq the property of accessibility:

```
Variable U:Type.
Variable R:U->U->Prop.

Inductive acc [R:U->U->Prop]:U->Prop :=
acc_intro : (x:U)((y:U)(R y x)->(acc R y))->(acc R x).

Definition wellfounded := [R:U->U->Prop]
(u:U)(acc R u).

Theorem wfa_wfp : (wellfounded R)
  ->(P:(U->Prop))((x:U)((y:U)(R y x)->(P y))->(P x))->(a:U)(P a).
Theorem wfp_wfa : ( (P:(U->Prop))((x:U)((y:U)(R y x)->(P y))->
  (P x))->(a:U)(P a))-> (wellfounded R).
```

Subsets. In a universe U , given $Y \subseteq U$ and $R \subseteq U \times U$, we define in Coq when the restriction of R to Y is well founded:

```
Require Ensembles.
Variable U:Type.
Variable R:U->U->Prop.
Variable Y:(Ensemble U).

Inductive Accss :U->Prop :=
accss_intro : (x:U)(In U Y x)->
               ((y:U)(In U Y y)->(R y x)->(Accss y))
               ->(Accss x).
```

```
Definition WF := (x:U)(In U Y x)->(Accss x).
```

Then, the following terms can be constructed:

```
Lemma Wfa_Wfp : WF -> (P:U->Prop)((x:U)(In U Y x)->
((y:U)(In U Y y)->(R y x)->(P y))->(P x))->
(a:U)(In U Y a)->(P a).
Lemma Wfp_Wfa:((P:U->Prop)((x:U)(In U Y x)->
((y:U)(In U Y y)->(R y x)->(P y))->(P x))->
(a:U)(In U Y a)->(P a)) ->WF.
```

In the particular case of (*Full_set* U), this concept is equivalent to the preceding *wellfounded*.

Let us call *wellfoundedInduction* to: $(P : U \rightarrow Prop)$

$$\left(\left((x : U) (Y x) \rightarrow \left((y : U) (Y y) \rightarrow (R y x) \rightarrow (P y) \right) \rightarrow (P x) \right) \rightarrow (a : U) (Y a) \rightarrow (P a) \right) \quad (1)$$

Orders. The module *Cpo* provides the basic facts about partial ordered sets. The constructor *P0* takes the carrier set, the relation, a proof that the carrier is not empty and a proof that the relation is an order and returns a partial order.

```
Record P0 : Type := Definition_of_P0 {
  Carrier_of: (Ensemble U);
  Rel_of: (Relation U);
  P0_cond1: (Inhabited U Carrier_of);
  P0_cond2: (Order U Rel_of) }.

Coq < Require Cpo.
Coq < Print P0.
Inductive PO [U : Type] : Type :=
  Definition_of_P0 : (Carrier_of:(Ensemble U); Rel_of:(Relation U))
                    (Inhabited U Carrier_of)
```

```

->(Order U Rel_of)
->(PO U)

```

```

Coq < Print Carrier_of.
Carrier_of =
[U:Type; p:(PO U)]
Cases p of (Definition_of_PO Carrier_of Rel_of _ _) => Carrier_of end
: (U:Type)(PO U)->(Ensemble U)

```

```

Coq < Print Rel_of.
Rel_of =
[U:Type; p:(PO U)]
Cases p of (Definition_of_PO Carrier_of Rel_of _ _) => Rel_of end
: (U:Type)(PO U)->(Relation U)

```

Also the predicates *greatest lower bound* and *least upper bound* are defined as inductive types:

```

Coq < Print Glb.
Inductive Glb [U : Type; D : (PO U); B : (Ensemble U); x : U]
: Prop :=
Glb_definition : (Lower_Bound U D B x)
->((y:U)(Lower_Bound U D B y)->(Rel_of U D y x))
->(Glb U D B x)

```

```

Coq < Print Lub.
Inductive Lub [U : Type; D : (PO U); B : (Ensemble U); x : U]
: Prop :=
Lub_definition : (Upper_Bound U D B x)
->((y:U)(Upper_Bound U D B y)->(Rel_of U D x y))
->(Lub U D B x)

```

3 The Results

Given a partial ordered set $(C, \leq)$, where $\leq$ represents a partial order relation in the carrier set C , and a function $F : C \rightarrow C$, we consider the subsets of C : $\{x \in C \mid x \leq F(x)\}$, set of postfixes or dense points, and $\{x \in C \mid F(x) \leq x\}$, set of prefixedes or closed points.

Theorem 1. (Tarski) *If F is monotone (i.e., $x \leq y$ implies $F(x) \leq F(y)$) and there exists the greatest lower bound (Glb) of $\{x \in C \mid F(x) \leq x\}$, then this is the least fixed point (lfp) of F .*

Dually, if F is monotone and there exists the least upper bound (Lub) of $\{x \in C \mid x \leq F(x)\}$, then this is the greatest fixed point (gfp) of F .

The representation in Coq goes as follows:

```

Variable U:Type.
Variable D:(PO U).
Local C := (Carrier_of U D).
Local R:= (Rel_of U D).

```

where D is the partially ordered set (C, R) with carrier C and relation R . Then defining:

Variable $F:U \rightarrow U$.

Definition $\text{monotone} := (x, y:U) (R \ x \ y) \rightarrow (R \ (F \ x) \ (F \ y))$.

Definition $\text{prefixed} := [x:U] (R \ (F \ x) \ x)$.

Definition $\text{postfixed} := [x:U] (R \ x \ (F \ x))$.

Axiom $\text{carr} : (x:U) (\text{In } U \ C \ x) \rightarrow (\text{In } U \ C \ (F \ x))$.

The axiom

$$\text{carr} \equiv \forall x \in U (x \in C \Rightarrow (F(x) \in C)) \quad (2)$$

ensures that F maps C into itself.

We prove in Coq:

Lemma $\text{Tarski\_prefixed} : \text{monotone} \rightarrow (x:U) (\text{Glb } U \ D \ \text{prefixed } x) \rightarrow (R \ (F \ x) \ x)$.

Lemma $\text{Tarski\_postfixed} : \text{monotone} \rightarrow (x:U) (\text{Lub } U \ D \ \text{postfixed } x) \rightarrow (R \ x \ (F \ x))$.

and, as a consequence:

Theorem $\text{Tarski\_pre\_fixed\_point} : \text{monotone}$

$$\rightarrow (x:U) (\text{Glb } U \ D \ \text{prefixed } x) \rightarrow x = (F \ x).$$

Theorem $\text{Tarski\_post\_fixed\_point} : \text{monotone}$

$$\rightarrow (x:U) (\text{Lub } U \ D \ \text{postfixed } x) \rightarrow x = (F \ x).$$

and

Lemma $\text{lwr\_fxd\_pt} : \text{monotone}$

$$\rightarrow (x, y:U) ((\text{Glb } U \ D \ \text{prefixed } x) / \wedge y = (F \ y)) \rightarrow (R \ x \ y).$$

Lemma $\text{gtr\_fxd\_pt} : \text{monotone}$

$$\rightarrow (x, y:U) ((\text{Lub } U \ D \ \text{postfixed } x) / \wedge y = (F \ y)) \rightarrow (R \ y \ x).$$

Given $x \in C$ let us consider the set

$$\text{postfixed}_x = \left\{ y \in C \mid \forall z \in U \left((\text{Lub} \{x, F(y)\} = z) \Rightarrow (y \leq z) \right) \right\}$$

Theorem 2. *Let $F : C \rightarrow C$ be a monotone function, $a = \text{Lub} \{u \in U \mid u \leq F(u)\}$ and $c_x = \text{Lub}(\text{postfixed}_x)$. Then $(x \leq a) \Rightarrow (x \leq F(c_x))$.*

Furthermore, if there exists $\text{Lub} \{x, F(y)\}$ for all y in postfixed_x , then $(x \leq F(c_x)) \Rightarrow (x \leq a)$.

in Coq:

Parameter $x:U$.

Definition $\text{postfixedx} := [y:U] (z:U) (\text{Lub } U \ D \ (\text{Couple } U \ x \ (F \ y)) \ z)$

$$\rightarrow (R \ y \ z).$$

Theorem $\text{kozen} : (\text{monotone } U \ D \ F) \rightarrow (a, c:U) (\text{Lub } U \ D$

$$(\text{postfixed } U \ D \ F) \ a) \rightarrow (\text{Lub } U \ D \ \text{postfixedx } c)$$

$$\rightarrow ((R \ x \ a) \leftrightarrow (R \ x \ (F \ c))).$$

In the construction of **kozen** we use the axiom:

Axiom $\text{ExistLubCouple} : (y:U) (\text{postfixedx } y)$

$$\rightarrow (\text{EXT } z:U \mid (\text{Lub } U \ D \ (\text{Couple } U \ x \ (F \ y)) \ z))$$

which ensures that exists the last upper bound of the set $\{x, F(y)\}$ for every $y \in \text{postfixed}_x$.

3.1 The Power-Set Case

In the particular case of $C = 2^U$ for a set U , with the inclusion order, the constant *carr* in (2) can be builded:

```
Variable U:Type.
Variable Family:(Ensemble (Ensemble U)).
Variable F:(Ensemble U)->(Ensemble U).
Definition intersec := [x:U](P:(Ensemble U))(Family P) -> (P x).
Definition uni := [x:U](EXT P:(Ensemble U) | (Family P) & (P x)).
Definition contained:=(Power_set_PO U (Full_set U)).
Local C := (Carrier_of (Ensemble U) contained).
Lemma carr:(P:(Ensemble U))(In (Ensemble U) C P)->
(In (Ensemble U) C (F P)).
```

Also, the axiom *ExistLubCouple* is trivially true in the power-set case.

```
Lemma ExistLubCouple: (x,y:(Ensemble U))
(postfixedx (Ensemble U) (contained U) F x y)
->(EXT z:(Ensemble U) |
(Lub (Ensemble U) (contained U)
(Couple (Ensemble U) x (F y)) z))
```

As a consequence, the previous results can be specialized to this case. Here, the *Lub* and *Glb* are given by *uni* and *intersec* respectively:

```
Lemma tent1:(Glb (Ensemble U) contained Family intersec).
Lemma tent2:(Lub (Ensemble U) contained Family uni).
```

and *lfp* and *gfp* are defined:

```
Definition lfp:=[x:U](P:(Ensemble U))(prefixed (Ensemble U) (contained U)
F P) -> (P x).
Definition gfp:=[x:U](EXT P:(Ensemble U) | (postfixed (Ensemble U)
(contained U) F P) & (P x)).
Lemma Glb_lfp:(Glb (Ensemble U) (contained U) (prefixed (Ensemble U)
(contained U) F) lfp).
Lemma Lub_gfp:(Lub (Ensemble U) (contained U) (postfixed (Ensemble U)
(contained U) F) gfp).
```

The principle of co-induction:

Theorem 3. *Let $F : 2^U \rightarrow 2^U$ be monotone and $gfp(F)$ its greatest fixed point. Then*

$$\forall P.(P \subseteq F(P) \Rightarrow P \subseteq gfp(F))$$

is just the application of

```
coind: (U:Type; D:(PO U); F:(U->U); x,y:U)
(postfixed U D F x) /\ (Lub U D (postfixed U D F) y)
->(Rel_of U D x y)
```

to the parameters *(Ensemble U)*, *contained*, *F*, *P* and *gfp*:

```

(postfixed (Ensemble U) contained F P)
/\(Lub (Ensemble U) contained
   (postfixed (Ensemble U) contained F) gfp)
->(Rel_of (Ensemble U) contained P gfp)

```

and then use `Lub_gfp`.

Dually, the principle of induction:

Theorem 4. *Let $F : 2^U \rightarrow 2^U$ be monotone and $\text{lfp}(F)$ its least fixed point. Then*

$$\forall P. (F(P) \subseteq P \Rightarrow \text{lfp}(F) \subseteq P)$$

The Reduction Lemma or Strong Principle of Co-induction

Theorem 5. (Reduction Lemma) *Let $F : 2^U \rightarrow 2^U$ be monotone and $\text{gfp}(F)$ its greatest fixed point. Then*

$$\forall P. P \subseteq \text{gfp}(F) \Leftrightarrow P \subseteq F(\text{gfp}(\lambda Q. (P \cup F(Q)))) \quad (3)$$

in Coq is a specialization of `kozen`:

```

Variable U:Type.
Variable P:(Ensemble U).
Variable F:(Ensemble U) -> (Ensemble U).
Local a:=(uni U (postfixed (Ensemble U) (contained U) F)).
Local c:=(uni U (postfixedx (Ensemble U) (contained U) F P)).
Local Lub1:=(tent2 U (postfixed (Ensemble U) (contained U) F)).
Local Lub1':=(tent2 U (postfixedx (Ensemble U) (contained U) F P)).
Definition G:=[Q:(Ensemble U)](Union U P (F Q)).

```

```

Theorem ReductionLemma: (monotone (Ensemble U) (contained U) F)
->(Rel_of (Ensemble U) (contained U) P (gfp U F))
<->(Rel_of (Ensemble U) (contained U) P (F (gfp U G))).

```

```

ReductionLemma < Proof.
ReductionLemma < Intros.
(Rel_of (Ensemble U) (contained U) P (gfp U F))
<->(Rel_of (Ensemble U) (contained U) P (F (gfp U G)))
ReductionLemma < Rewrite <- agfp.

```

use the `Rewrite` tactic to replace `(gfp U F)` by `a`, because there is already a term `agfp` with type `a==(gfp U F)` gives the following subgoal with the same environment and context:

```

(Rel_of (Ensemble U) (contained U) P a)
<->(Rel_of (Ensemble U) (contained U) P (F (gfp U G)))

```

```
ReductionLemma < Rewrite foo.
```

replace $(\text{gfp } U \text{ } G)$ by c by using the term `foo` with type $(\text{gfp } U \text{ } G) == c$ produces the goal:

```
(Rel_of (Ensemble U) (contained U) P a)
<->(Rel_of (Ensemble U) (contained U) P (F c))
```

finally, apply `kozen`:

```
ReductionLemma < Apply (kozen (Ensemble U)
(contained U) F P H a c Lub1 Lub1').
Subtree proved!
```

Winskel uses this lemma with a finite set of states (more precisely, when P is a singleton) to define a *model checking* algorithm. Using a relation which makes the right-hand side of (3) simpler to verify and, because the set of states is finite, this relation turns out to be well-founded ensuring termination of the model checking algorithm. But if we consider systems with an infinite number of states termination will no longer be guaranteed in this manner.

Of course we have the dual of the Reduction Lemma:

Corollary 1. *Let $F : 2^U \rightarrow 2^U$ be monotone and $\text{lfp}(F)$ its least fixed point. Then*

$$\forall P. \text{lfp}(F) \subseteq P \Leftrightarrow F(\text{lfp}(\lambda Q. (P \cap F(Q)))) \subseteq P.$$

but this is not a very useful result because what we need is to obtain some sufficient condition to ensure that $P \subseteq \text{lfp}(F)$.

The next result[1], however, provides such a condition.

Well-founded Induction on Minimum Fixed-points. Given a set $Y \subseteq U$, a *covering* of Y is a family of sets $\{X_n\}_{n \in C}$ such that $Y \subseteq \bigcup_{n \in C} X_n$ for some index set C . Also, for every $X \subseteq U$, let us denote $(\prec X)$ the set $\{y \in Y \mid \forall x \in X. y \prec x\}$.

Theorem 6. *If $F : 2^U \rightarrow 2^U$ is a monotone function and $Y \subseteq U$, has a well-founded relation $\prec \subseteq Y \times Y$ and a covering $\{X_n\}_{n \in C}$, then*

$$\forall n \in C \ X_n \subseteq F(\text{lfp}(\lambda Q. (\prec X_n) \cup F(Q))) \Rightarrow Y \subseteq \text{lfp}(F)$$

In order to construct a proof in `Coq`, we use the representation given in subsection **Subsets** of section (2.1) and the term (1). Also, it is necessary to require the facts about fixed points already proved.

```
Variable U:Type.
Variable C:Set.
Variable F:(Ensemble U)->(Ensemble U).
Variable X:C-> (Ensemble U).
Variable R:U->U->Prop.
Variable Y:(Ensemble U).
```

in this context, we represent $(\prec W)$:

```
Definition less_than := [W:(Ensemble U)]
                        [x:U]
                        (In U Y x) /\ (w:U) (In U W w) -> (R x w).
```

now we define the collection of functions $\lambda Q. (\prec X_n) \cup F(Q)$:

```
Definition GT := [n:C] [Q:(Ensemble U)] (Union U (less_than (X n)) (F Q)).
```

and to codify the hypothesis $Y \subseteq \bigcup_{n \in C} X_n$:

```
Hypothesis A1: (x:U) (In U Y x) -> {n:C | (In U (X n) x)}.
```

Then, the theorem can be stated and proved in Coq:

```
Theorem WFIMFP : (monotone (Ensemble U) (contained U) F) -> WF ->
((n:C) (Rel_of (Ensemble U) (contenido U) (X n) (F (lfp U (GT n)))) ->
(Rel_of (Ensemble U) (contenido U) Y (lfp U F))).
```

As pointed out in [1] the reciprocal is also true. Just take $I = \{1\}$, $U_1 = U$, and $\prec$ the empty relation.

4 Conclusions and Future Work

We obtain proof objects for some facts which are basic to the implementation of the modal μ -calculus in a logical framework. This work is part of a large project from the authors devoted to model in Coq both a concurrent system and its logical properties. But beside its practical and theoretical motivations, this work can give some insights on the expressive power of Coq or similar theorem proving tools. The proofs have been generalized as much as possible in order to have a more useful library.

Complete Coq code is available at <http://lfcia.org/publications.shtml>

References

1. Andersen, H. R.: On Model Checking Infinite-State Systems. Proceedings of LFCS'94. LNCS 813. Springer-Verlag. pp. 8–17. (1995)
2. Collins, G.: A Proof Tool for Reasoning about Functional Programs. Proc. of TPHOL'96, Turku, Finland (1996)
3. Constable, R. L., et al: Implementing Mathematics with the NuPRL Proof Development System. Prentice Hall (1986)
4. Dowek, G., Felty, A., Herbelin, H., Huet, G., Murty, Ch., Parent, C., Paulin-Mohring, Ch., Werner, B.: The Coq proof assistant user's guide. INRIA Technical Report 134. (1993)
5. Freire, J. L., Freire, J. E., Blanco-Ferro, A., Sánchez, J. J.: Fusion in Coq. LNCS 2178. Springer (2001)
6. Gordon, A. D. : Bisimilarity as a Theory of Functional Programming. Technical Report NS-95-3, Basic Research in Computer Science, University of Aarhus, (1995).

7. Gordon, M. J. C., Melham, T. F.: *Introduction to HOL: A Theorem-proving Environment for Higher-Order Logic*. Cambridge University Press (1993)
8. Kozen, D.: Results on the propositional μ -calculus. *Theoretical Computer Science*, 27, pp. 333–354, (1983)
9. Gordon, M. J. C., Milner, R., Wadsworth, Ch. P.: *Edinburgh LCF: A Mechanised Logic of Computation*. Springer LNCS 78 (1979)
10. Luo, Z., Pollack, R.: *LEGO Proof Development System: User's Manual*. LFCS Report ECS-LFCS-92-211, Departament of Computer Science, University of Edinburgh (1992)
11. Miculan, M.: On the formalization of the modal μ -calculus in the Calculus of Inductive Constructions. *Information and Computation*, Vol. 164, No. 1, pp. 199–231, (2001)
12. Owre, S., Rushby, J. M., Shankar, N.: *PVS: A prototype verification system*. In Deepack Kapur, editor, 11th International Conference on Automated Deduction (CADE), LNCS 607, Springer-Verlag (1981)
13. Sprenger, Ch.: A Verified Model Checker for the modal μ -calculus in Coq. TACAS '98, Lisbon, Portugal, Springer LNCS 1384, pp. 167–183, (1998)
14. Tarski, A.: A Lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5, pp. 285–309, (1955)
15. Verma, K. N.: *Reflecting Symbolic Model Checking in Coq*. INRIA. (2000)
16. Walukiewicz, I.: Notes on the Propositional μ -calculus: Completeness and Related Results. BRICS Notes Series NS-95-1. ISSN 0909-3206. (1995)
17. Winskel, G.: A note on model checking the modal ν -calculus. In G. Ausiello, M. Denazi-Ciancaglini and S. Rochi Della Rocca, editors, Springer LNCS 372 (1989)
18. Yu, Shen-Wei: *Formal Verification of Concurrent Programs Based on Type Theory*. Ph.D. Thesis, Departament of Computer Science. University of Durham (1998)

A Throttle and Brake Fuzzy Controller: Towards the Automatic Car

J.E. Naranjo, J.Reviejo, C. González, R. García, and T. de Pedro

Instituto de Automática Industrial, CSIC
La Poveda, 28500 Arganda del Rey, Madrid, Spain
Tlf. 34 91 8711900, Fax 34 91 8717050

{jnaranjo,jesus,gonzalez,ricardo,tere}@iai.csic.es
<http://www.iai.csic.es/autopia>

Abstract. It is known that the techniques under the topic of Soft Computing have a strong capability of learning and cognition, as well as a good tolerance to uncertainty and imprecision. Due to these properties they can be applied successfully to Intelligent Vehicle Systems. In particular Fuzzy Logic is very adequate to build qualitative (or linguistic) models, of many kinds of systems without an extensive knowledge of their mathematical models. The throttle and brake pedal and steering wheel, are the most important actuators for driving. The aim of this paper is to integrate in a qualitative model the vehicle operation and the driver behavior in such a way that an unmanned guiding system can be developed around it [1] [2]. The automation of both pedals permits to direct the speed control from a computer and so, to automate several driving functions such as speed adaption, emergency brake, optimum speed selection, safe headway maintenance, etc. The building and design of fuzzy controllers for automatic driving is based on the drivers' know-how and experience and the study of their behavior in maneuvers. The use of fuzzy controllers allows achieving a human like vehicle operation. The results of this research show a good performance of fuzzy controllers that behave in a very human way, adding the precision data from a DGPS source, and the safety of a driver without human lacks such as tiredness, sensorial defects or aggressive standings.

1 Introduction

The development of Intelligent Transport Systems (ITS) generates a new broad scope of technologies and systems aiming to the automation of vehicles in the first place and to the automation of the traffic management in the second place. Many social benefits can be expected from ITS, among them improving the passenger comfort and safety, increasing the road exploitation and decreasing the environmental pollution and the number and gravity of accidents [3]. For the sake of this paper the scope of ITS is limited to the vehicle automation, namely to software and hardware engines embedded in the mass-produced vehicles.

Some of the most popular automation engines are Cruise Control (CC) and Adaptive Cruise Control (ACC). The CC, already included in some serial cars, allows keeping a target velocity, pre-selected by the driver, automatically. The ACC, present only in prototypes, allows adapting automatically the velocity of the car to the



Fig. 1. AUTOPIA testbed automatic vans

conditions of roads and traffic [4], what is achieved by calculating a velocity profile that maintains a safe gap with the precedent vehicle. Until now the existing CC's and ACC's operate upon the accelerator only, leaving the braking operation to driving aid systems that alert the driver in case of emergency only, when the car has to be stopped immediately [5].

The name AUTOPIA encloses a set of projects aiming eventually towards automatic driving by implementing artificial pilots. Our approach is to develop artificial pilots that mimic the qualitative perceptions and actions of human pilots [1] [6] and operate the controls of the car -the throttle, the brake and the steering wheel. The key parts of these pilots are fuzzy logic procedures [7] that interpret qualitatively the data provided by sensors and control [8] the steering and the speed [9].

Two Citroën Berlingo vans (Figure 1) have been equipped as test-bed cars in order to probe our artificial pilots. Each vehicle has been instrumented with an industrial on board PC-based computer, electronic interfaces and two electric motors, one for the steering wheel and another for the brake pedal. The throttle automation is done by an electronic interface with the on board computer [10].

2 A Joint Throttle and Brake Control for a Human Like Driving

As it has been said, there are two kinds of controllers for the speed of the vehicles, also called longitudinal controllers, the CC and the ACC [11]. Both control the velocity by means of the accelerator pedal only. The speed maintenance performed with the throttle is effective at high speeds only, and it is the classical function of the cruise control. Nevertheless people drive cars operating upon the throttle or the brake

alternatively. Taking into consideration that one guideline of the AUTOPIA program is to imitate the human behavior in the driving process, the controller explained in this paper [2] has the capability of performing a human-like control of the throttle&brake pedal set. As other of our precedent controllers, also this is based on fuzzy logic and tested in real experiments, not simulations.

A first requirement of the AUTOPIA guideline is that the throttle&brake set control be formed by two different controllers, one for the brake and other for throttle, but they must work in a cooperative way; in other words, they can act independently, but both have the same goal: to maintain the adequate velocity profile. Another design requirement is that, as in human drivers, only one controller is enabled at any moment, but the transition between them must be soft, offering a comfortable cruise as good as good human drivers do. Passenger comfort is the most important performance evaluation factor, because people are the final users and main beneficiaries of ITS products [12].

Finally a functional consideration in order to design the control system is that, since the throttle and the brake commands have the same task (both can speed up and down the car independently) both controllers are probably going to be very similar.

Nevertheless for clarity the part of the throttle and the part of the brake of the conjoint control system are exposed separately

2.1 The Throttle Control Functions

Reviewing the use that humans do of the throttle, we may define a set of functions of the control system. These functions correspond to the intuitive uses of this pedal:

1. Strong speed increasing
2. Speed maintenance
3. Slight speed reducing

The first function represents the natural effect of a throttle. The main target of this pedal is to boost significantly the speed of the car. When we ask a human driver about the throttle use the first answer is this: to augment the speed.

But this function is not enough to reproduce all the effects that a human driver wants to produce when he uses this pedal. In the second function, maintaining the speed, two cases have to be considered, current speed is higher than the target speed and current speed is lower than the target speed. In the first, the driver must step a little off the throttle in order to decelerate the car; in the second case the control action is to step softly on the pedal for increasing tenderly the velocity. The aim of both cases is a fine adjusting to the cruise desired speed, and the control action must always be soft.

Contrary to the first function, the third of them, the action of decreasing the target speed by means of the throttle is not natural at first sight. Nevertheless this function has to be included to reproduce some human maneuvers –colloquially known as motor braking- in which human drivers step off the throttle to produce a slight and slow braking effect. It is used when the distance for braking is long or a soft braking is enough. This function is similar to the second but we have separated it because the objective is different. One reduces a little the speed for maintaining the target, while attempting the deceleration to be small, and the other reduces the speed slowly, but quantitatively in a considerable amount, the acceleration or deceleration notwithstanding. In any case you have to stop accelerating before start braking.

2.2 The Brake Control Functions

As in the throttle, we have defined the main brake pedal functions in order to complete the definition of the speed control system. A human driver uses the brake to perform the intuitive operations defined below:

1. Strong speed decreasing
2. Speed maintenance
3. Slight speed increasing

The first function is the natural intuitive effect of a brake system, hard braking until stopping the car if needed. A human driver uses this pedal to stop or reduce strongly the speed of the car, to answer to emergency situations as well as to reduce the current speed at the desired moment, adapting the speed to the traffic or the lane characteristics, independently of inertia of the car or the orography of the road.

But, as before, this function does not reproduce all the effects that a human driver wants to produce when he steps on a car brake. The second function, the maintenance of a target speed, generates an advanced cruise control when it is combined with the above mentioned throttle controller. There are also two cases that have to be considered: the current speed is lower than the target speed and the current speed is higher than the target speed. In the first, the action that must be performed by the brake control is to step off the pedal and wait for a speed increment that may be caused by the throttle or the characteristics of the road, downhill for instance. In the second case the effect of control action is decreasing the velocity and it is used in this cruise control to reduce the acceleration if the current speed is higher than the target. The addition of the brake control to the traditional cruise control allows performing an automatic speed control at low speeds, as it happens in city driving and extends the uses of CC beyond its actual limits.

As in the throttle control, the action of increasing the target speed by means of the brake has to be included to reproduce the maneuvers of de-breaking, which are necessary in downhill tracks or mountain roads. In these stages a car can increase significantly its speed when the brake pedal is stepped off, without pressing the throttle. In these cases, the driver can perform a full cruise control only stepping the brake pedal. This function is important at time of control definition.

3 Fuzzy Control Procedures

As we have already said two independent fuzzy controllers, for accelerating and breaking, make up the pilot. Formally both controllers are very similar; they have the same rules, the same input variables and dual output variables. Even more, the labels of the linguistic values of the variables are the same. The only difference is that the membership functions of these linguistic values are different for each controller.

To fix ideas, it is convenient to remind briefly the principal formal and functional elements of a fuzzy controller [13]. The core of the base of knowledge of a fuzzy controller is a set of *if ... then* rules in which the propositions of both sides, antecedents in the left and consequents in the right, are fuzzy. The variables involved in the propositions are also fuzzy, or linguistic, what means that they can take linguistic values, or fuzzy predicates. The variables of the antecedents are the control inputs and the variables of the consequents are the control outputs. Finally the

meaning of the linguistic values –also named as fuzzy partitions- is defined for each process. Associating possibility functions, called membership functions, to the linguistic values, does this. In this paper the definition of fuzzy variable is noted by its name and a set of labels, standing for their linguistic values, enclosed between { }. Functionally in each control cycle a fuzzy controller proceeds according to the following list of actions:

- Acquire the numerical values of the control inputs,
- Compute the degrees of compatibility between the numerical values and their correspondent linguistic values,
- Compute the inference of every rule. This yields the degree in which every rule contributes to the control action,
- Compute the total control action by means of the defuzzification process.

A last observation is that the fuzzy controller is based on our own fuzzy software, ORBEX [14], more powerful than the usual fuzzy tools. In particular the antecedents of the rules can contain fuzzy modifiers, such as LESS THAN, MORE THAN and others. As the modifiers allow deriving new linguistic values from the fuzzy values included in the definitions of the variables, the number of linguistic values and the number of rules can be reduced. Besides the expressions of the rules are close to the natural language.

3.1 Fuzzy Throttle Controller

The base of knowledge of the fuzzy throttle controller, formed by the inputs, outputs and rules is detailed next. To establish a parallelism with classical controllers, it can be said that it is a fuzzy PD controller, with a proportional part, the speed error, and a derivative part, the acceleration.

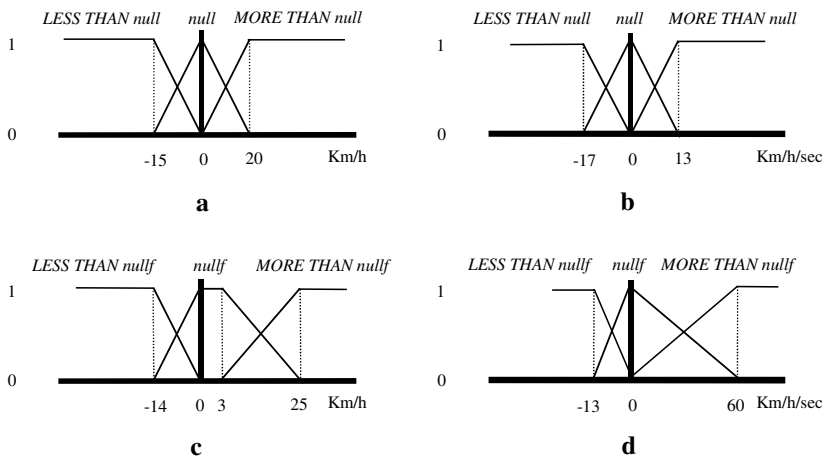


Fig. 2. Membership functions shapes for the input variables of the throttle and brake controllers. Figure a, *Speed_error* for the throttle. Figure b, *Acceleration* for the throttle. Figure c, *Speed_error* for the brake pedal. Figure d, *Acceleration* for the brake pedal.

Input variables: There are two inputs named *Speed_error* and *Acceleration*, both have a unique linguistic value, named *null*. Other linguistic values can be derived by the application of modifiers MORE THAN and LESS THAN. It is important to notice that, though formally the linguistic values of the variables are identical, semantically are different, because the shape of the membership function attached to the value *null* for each one gives the meaning.

Speed_error {null}

The membership function shape for the *Speed_error* variable is represented in Fig. 2. a. This shape means that, when the speed error is negative, the speed of the car is lower than the target speed, and less than -15 Km/h, the function value is 0. When the speed error is between -15 Km/h and 0 Km/h the function increases from 0 to 1. Similarly if the speed of the car is faster than the target speed, the speed error is positive. When the error increases from 0 Km/h to 20 Km/h, the function value decreases from 1 to 0.

Acceleration {null}

The membership function shape for *Acceleration* variable is shown in Fig. 2. b

The shape of the membership function *null* for the acceleration is interpreted as follows. If the acceleration of the car is negative and its value increases from -17 Km/h/sec to 0 Km/h/sec the function value increases too from 0 to 1. If the acceleration is positive, the function value decreases from 1 to 0 when the acceleration increases from 0 Km/h/sec to 13 Km/h/sec. The use of Kilometer/hour/seconds is due to the fact that Km/h is the unit of the car's tachometers and intuitively a driver calculates the acceleration of the car watching the tachometer variations per time unit, in our case seconds.

Throttle {up, down}

Output variable: There is only one output variable named *Throttle* that can take two linguistic values *up* and *down*. To model properly the acceleration effects, these linguistic values have attached singletons as membership functions.

Finally the fuzzy control rules for the throttle are:

IF *Speed_error* MORE THAN *null* THEN *Throttle up*
 IF *Speed_error* LESS THAN *null* THEN *Throttle down*
 IF *Acceleration* MORE THAN *null* THEN *Throttle up*
 IF *Acceleration* LESS THAN *null* THEN *Throttle down*

Where we have marked in upper case letter the ORBEX modifiers. The names of the input and output variables are marked as italic and the labels of the variables are marked as bold.

It must be noted that the rules of the speed error refer to the proportional part of control. Both rules work cooperatively, stepping on the throttle when the current speed is lower than the target one and stepping off it when the current speed is higher than the desired.

The acceleration rules build the derivative part of the control. Its function is to smooth the proportional rules actuation. This way, if the car is increasing speed, the first acceleration rule lightens the pressure over the pedal in order to have a better

comfort and a soft adaption to the target speed. If the car is decelerating only with the throttle (motor braking) the second rule allows a soft decreasing for adapting to the desired speed.

3.2 Fuzzy Brake Controller

Analogously to the throttle controller, the base of knowledge of the brake controller is written next. It has the same control inputs with the same definitions as the fuzzy PD controller. The control output is named *Brake* and is defined as the *Throttle*, with equal linguistic values and membership functions.

Input variables: there are the same two input variables that for the throttle: *Speed_error* and *Acceleration*, which membership functions are shown respectively in Fig. 2.c. and Fig. 2.d. They have a unique linguistic value, *nullf*, semantically different for each one, that can be operated also by the MORE THAN and LESS THAN modifiers.

Speed_error {nullf}

The shape of this membership function defines the actuation over the brake pedal of the proportional part of the fuzzy PD controller when the speed is lower than the target (negative) or when it is higher than it (positive). A difference with the throttle *Speed_error* membership function is that the function for the brake is trapezoidal. For values of speed error between 0 Km/h to 3 Km/h, the value of this function is 1. This is for the control to allow an error of 3 Km/h on the target speed so as not to use a powerful actuator, the brake pedal, for little corrections (as human drivers do). This definition avoids overactuations and leaves the little brake operation to the motor braking.

Acceleration {nullf}

The appearance of this membership function is triangular as in the throttle, but very asymmetrical.

Brake {up, down}

Output variable: There is only one output variable named *Brake* that can take two linguistic values *up* and *down*. To model properly the braking effects, these linguistic values have attached singletons as membership functions.

The fuzzy control rules for the brake are:

IF *Speed_error* MORE THAN *nullf* THEN *brake down*

IF *Speed_error* LESS THAN *nullf* THEN *brake up*

IF *Acceleration* LESS THAN *nullf* THEN *brake up*

In this case, the rules are only three and the first two are also similar to the throttle ones. These rules try to step on the pedal when the speed is higher than the target, and step off it when the speed is lower. It can be noticed that there is only one derivative rule for this controller, in effect when the car is braking, the acceleration rule increases the pressure on the brake pedal in order to perform a soft fit to the desired speed. The other derivative rule has been suppressed because it would work when the control for the brake has to accelerate and this must not happen. The car must fit to the user defined target speed softly only with the throttle, without brake intervention.

3.3 Fuzzy Throttle and Brake Controller

The experiments in the following section will show that the pilot provides a comfortable cruise and a fast and soft speed adaptation. In effect, the pilot has to dispatch the brake control or the throttle control to adapt the cruise car to the traffic conditions. Our pilot assures a smooth transition between throttle and the brake controllers. This is achieved by means good definitions for the brake of the membership functions as we explain next.

Comparing the definitions of the membership functions for the *null* and *nullf* values for the Speed_error we do the following observations: a) the left parameter of the *null* value is -15 km/h and the same parameter for the *nullf* value is -14 km/h; this little difference assures that the action of stepping off the brake starts before the action of stepping on the throttle, when the current speed is less than the target one. b) the right parameter of the *null* value is -20 km/h and the same parameter for the *nullf* value is -25 km/h; thus, for the same current value of the Speed_error, it is assured that the action of stepping off the throttle is stronger than the action of stepping on the brake, when the current speed is bigger than the target one. c) finally the summit of the *nullf* definition, between 0 and 3 km/h, is introduced to take in account the maximum error permitted before starting some brake actuation.

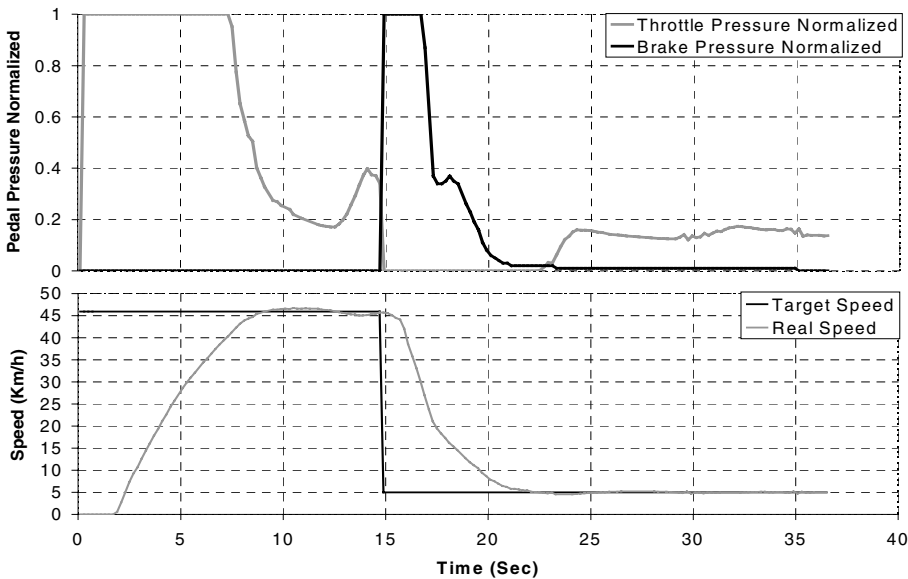


Fig. 3. The experiment starts with the car stopped and the target speed fixed at 46 Km/h. The control operation is to press fully on the throttle pedal in order to attain this desired speed. As the car is increasing its speed, the control pressure over the throttle decreases for a soft target adaption. After some seconds of speed maintenance, the target speed is set at 5 Km/h. Then the throttle is fully released and the brake pedal is pressed in order to reduce the speed as soon as possible in a human like way; first stronger and, after a hard reduction, step slow off the pedal for a soft adaption. At the end of the experiment the speed is maintained at 5 Km/h.

4 Experiments

We are now to comment the experimental graphics showing speed adaptation. All of them are divided in two panels. The upper shows the behavior of the throttle and brake pedal control actuations, normalized between 0, for pedal full up, and 1 for pedal full down. The bottom panel represents the target speed and real speed profiles obtained with the combined actuation of both pedals, measured in Kilometers per hour. Though we have reached automatic driving test up to 80 Km/h, in this paper we show automatic speed tracking until 55 Km/h. This is about the maximum allowed speed for urban driving, one of the aims of our research. A first observation is that the maximum speed error on the stationary state is lower than 3 km/h, which is the limit set into the controllers and it is a lower limit than what human drivers do.

The first graphic (Fig. 3.) shows the car starts moving being the target speed fixed; when the limit is reached the target speed is suddenly decreased. As it is expected the speed is reached controlling only the throttle without acting on the brake pedal. The throttle actuation curve shows the maneuvers to maintain speed. Actuation on the throttle is suppressed when the target speed is decreased, and brake pedal actuation starts. When the lower speed is reached actuation on the brake stops and the speed is maintained with the throttle. Let us note here that this takes place on a flat road. Fig. 4. shows a similar performance in two consecutive target speed decreasing steps.

The third graphic (Fig. 5) is similar to the previous one, but it has two parts. The first one is downhill and the second one uphill. We see that it is more difficult to maintain speed downhill and, even if the slope is not great enough to require break intervention, the control is not as good as in the first case, although the 3 Km/h limit is not surpassed.

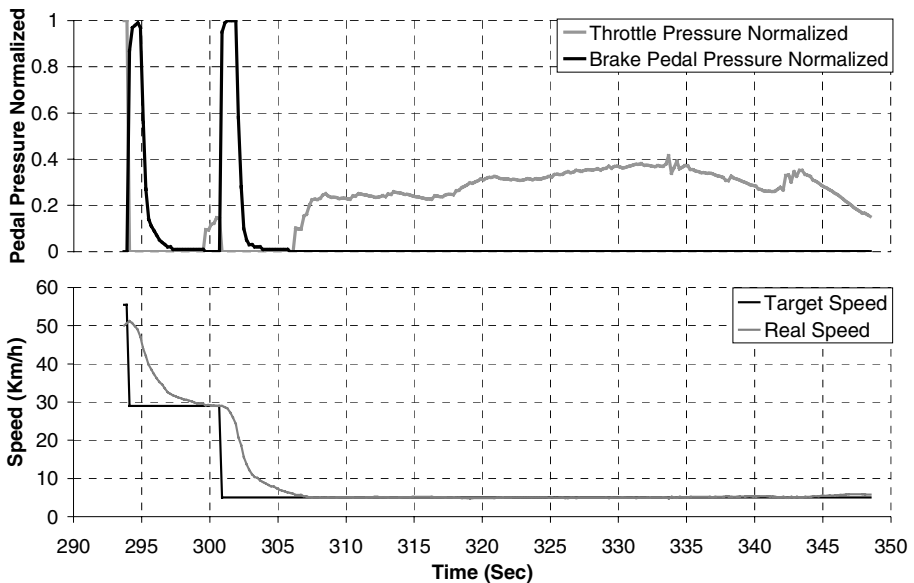


Fig. 4. This figure shows the conjoint throttle&brake pedal control performance in two consecutive target speed reductions. First the car is circulating about 50 Km/h and the target speed is set to 29 Km/h. The control actuates over the brake and throttle in order to fit at this speed. When it is reached, a new desired speed is defined: 5 Km/h. The controller activates the actuators again and adjusts to the new speed in a human like way.

5 Conclusions

As conclusion we can say that this integration of the throttle&brake pedal set control through a fuzzy controller is a new and novel application of fuzzy control and soft computing, developing new algorithms and models. The results of the control experiments show a good performance in a near human way driving, with safety and comfortability.

This computational solution and software architecture are different and as reliable as other research on ITS in the world.

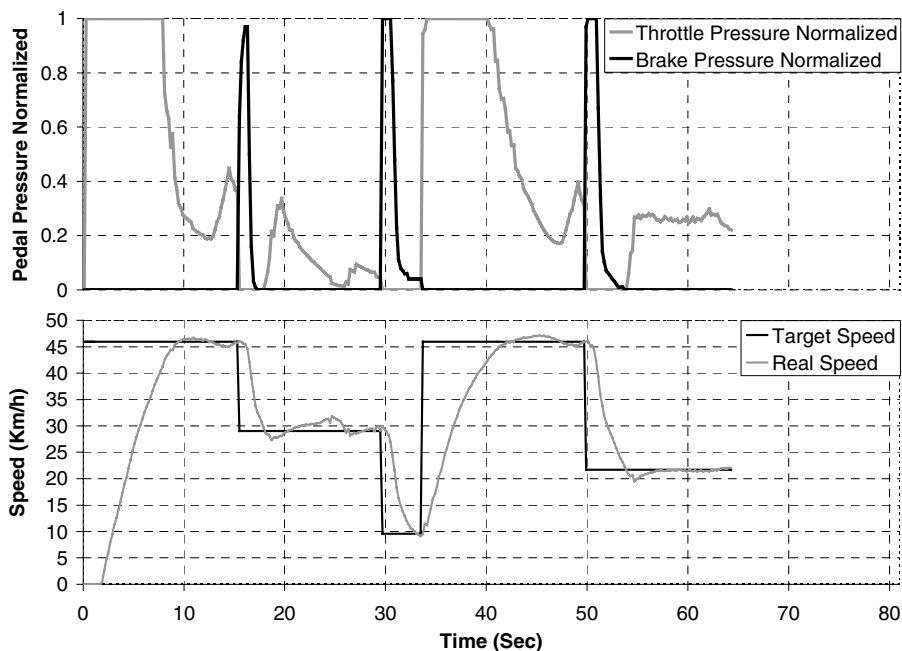


Fig. 5. Several speed adaptations and actuators response at downhill and uphill roads.

Acknowledgements. This work is part of the project ISAAC, DPI2002-04064-C05-02 of the Spanish Ministry of Science and Technology and project COPOS, Ministerio de Fomento BOE 280, November 22, Res. 22778.

References

1. Huang, S. , Ren, W.: Use of neural fuzzy networks with mixed genetic/gradient algorithm in automatic vehicle control. IEEE Transactions on Industrial Electronics, 46, (1999) 1090–1102.
2. García, R., de Pedro, T.: First application of the Orbex coprocessor: Control of unmanned vehicles. Mathware and Soft Computing n7, vol 2–3 (2000) 265–273.

3. STARDUST: Scenarios and Evaluation Framework for City Case Studies. European Comission Fifth Framework Programme Energy, Environment and Sustainable Development Programme Key Action 4: City of Tomorrow and Cultural Heritage, Deliverable 2, 3, (2002).
4. Jones, W.D.: Keeping Cars from Crashing. *IEEE Spectrum*, September 2001, 40–45.
5. Holve, R., Protzel, P., Naab, K.: Generating Fuzzy Rules for the Acceleration Control of an Adaptive Cruise Control System. *Fuzzy Information Processing Society 1996 NAFIPS. Biennal Conference of the North American* (1996) 451–455.
6. García, R., de Pedro, T.: Automatic Car Drivers. 31st Intern. Symposium on Automotive Technology and Automation, June 1998.
7. Zadeh, L.A.: Fuzzy Sets. *Information and Control* 8 (1965) 338–353.
8. Sugeno, M. et al.: Fuzzy Algorithmic Control of a Model Car by Oral Instructions. In: Hirota, K. and Yamakawa, T. (eds.): *IFSA'87 special issue on fuzzy control*, October 1987.
9. García, R., de Pedro, T., Naranjo, J.E., Reviejo, J., González, C.: Frontal and lateral control for unmanned vehicles in urban tracks. *IEEE Intelligent Vehicle Symposium*, June 2002.
10. Alcalde, S.: Instrumentación de un vehículo eléctrico para una conducción automática. Degree thesis, Escuela Universitaria de Informática, Universidad Politécnica de Madrid, January 2000.
11. Ioannou, P.A., Chien, C.C.: Autonomous Intelligent Cruise Control. *IEEE Transactions on Vehicular Technology*, volume 42 (1993) 657–672.
12. Crosse, J.: Tomorrow's World. *Automotive World*, January/February 2000, 46–48.
13. Klir, G.J., St Clair, U.H., Yuan, B.: *Fuzzy Set Theory: Foundations and Applications*. Pearson Education POD (1997).
14. García, R., de Pedro, T.: Modeling Fuzzy Coprocessors and its Programming Language. *Mathware and Soft Computing*, n. 5 (1995) 167–174.

ADVOCATE II: ADVanced On-Board Diagnosis and Control of Autonomous Systems II

Miguel Angel Sotelo¹, Luis Miguel Bergasa¹, Ramón Flores¹, Manuel Ocaña¹, Marie-Hélène Doussin², Luis Magdalena³, Joerg Kalwa⁴, Anders L. Madsen⁵, Michel Perrier⁶, Damien Roland⁷, and Pietro Corigliano⁸

¹ University of Alcalá, Escuela Politécnica. Campus Universitario s/n, Alcalá de Henares, 28871, Madrid, SPAIN.

michael@depeca.uah.es,

<http://www.depeca.uah.es>

² GETRONICS, Europarc bat D,

Technopôle de Chateau-Gombert, 13013 Marseille (France)

³ ETSI Telecomunicación, Universidad Politécnica de Madrid (UPM)
Madrid 28040 (Spain)

⁴ STN-ATLAS Elektronik GmbH,

Sebaldsbrucker Heerstraße 235, 28305 Bremen (Germany)

⁵ Hugin Expert A/S,

Niels Jernes Vej 10, 9220 Aalborg (Denmark)

⁶ IFREMER, Zone portuaire de Bregailon, BP 330, 83597 La seyne-sur-mer (France)

⁷ E-MOTIVE, Marseille (France)

⁸ INNOVA S.p.A., Via della Scrofa 117, 00186 Rome (Italy)

Abstract. A way to improve the reliability and to reduce costs in autonomous robots is to add intelligence to on-board diagnosis and control systems to avoid expensive hardware redundancy and inopportune mission abortion. According to this, the main goal of the ADVOCATE II project is to adapt legacy piloting software around a generic SOAP (Simple Object Access Protocol) architecture on which intelligent modules could be plugged. Artificial Intelligent (AI) modules using Belief Bayesian Networks (BBN), Neuro-Symbolic Systems (NSS), and Fuzzy Logic (FL) are coordinated to help the operator or piloting system manage fault detection, risk assessment, and recovery plans. In this paper, the specification of the ADVOCATE II system is presented.

1 The ADVOCATE II Architecture

ADVOCATE II introduces intelligent techniques for diagnosis, recovery and re-planning into UUVs (Unmanned underwater vehicles) and UGVs (Unmanned Ground Vehicles). The global objective of the project is to enhance the level of reliability and efficiency of autonomous robotic systems, as described below:

- To construct an open, modular, and generic software architecture for autonomous robotic systems diagnosis and control.

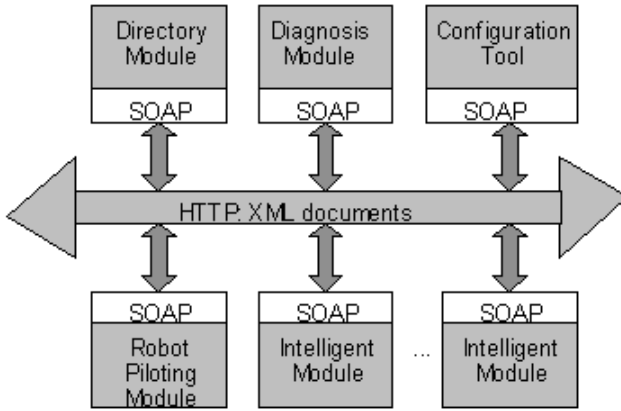


Fig. 1. The ADVOCATE II Architecture

- To develop or improve a set of intelligent diagnosis modules fully compatible with this architecture and tested in operational applications.
- To carry out practical tests and demonstrations on a set of operational prototypes in order to prove operability and efficiency of this solution in several application fields, and particularly for Autonomous Underwater Vehicles (AUVs) and Autonomous Ground Vehicles (AGVs).

ADVOCATE II is based on a distributed architecture, and a generic protocol (SOAP/XML technology implementing HTTP) for communication between the different modules. The ADVOCATE II architecture is distributed around a SOAP bus as depicted in figure 1. The architecture is modular, easy to evolve and to adapt to legacy piloting systems [1] [2]. It comprises five different types of modules.

1.1 Robot Piloting Module (RPM)

This module manages the mission plans and communicates directly with the vehicle sensors and actuators. Several RPMs, each of them working on a specific subsystem, can be plugged onto the ADVOCATE II architecture. Each end-user participating in the project (UAH, STN ATLAS, and IFREMER) is responsible for the corresponding piloting modules. A brief description of the different Robotic platforms used in the ADVOCATE II project is provided below.

IFREMER Vehicle Platform and Robot Piloting Module. The application deployed by IFREMER is based on an experimental underwater vehicle, called VORTEX, operated in a test pool or in simulation. The vehicle is a small experimental Remotely Operated Vehicle (ROV), but it can be considered as an AUV from a control point of view since fully automatic missions can be programmed and performed. VORTEX can be considered as a vehicle that belongs to the class of working AUVs, not dedicated to long survey tasks, but to



Fig. 2. VORTEX vehicle

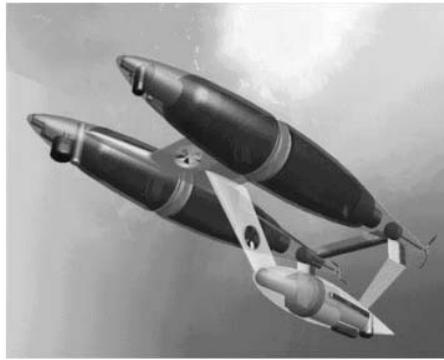


Fig. 3. DeepC Vehicle

intervention tasks (offshore application for instance). Mechanically, the vehicle structure consists of a basis tubular structure on which the different actuators are arranged, without pre-defined locations, as depicted in figure 2. Central to this structure is the main electronics package containing the vehicle electronics as well as the different set of sensors : attitude, depthmeter, gyrometers, video camera, sonar and sounders.

The control architecture of VORTEX is located on a VME-based system, reachable through Ethernet, where different software modules can be connected to the vehicle controller:

- Man-Machine Interface used to supervise and operate the vehicle.
- Mission Manager used to program simple or complex missions.

STN-ATLAS Vehicle Platform and Robot Piloting Module The DeepC vehicle developed under the support and promotion of the Federal Ministry of Education and Research of Germany is a fully autonomous underwater vehicle (AUV), depicted in figure 3, with the related components on the water's surface for oceanographic and oceanologic applications.

One of the outstanding features of the AUV is the “reactive autonomy”. This property allows situation-adapted mission and vehicle control on the basis of



Fig. 4. UAH platform: BART (Basic Agent for Robotic Tasks) robot.

multi-sensor data fusion, image evaluation and higher-level decision techniques. The aim of the active and reactive process is to achieve high levels of reliability and safety for longer underwater missions in different sea areas and in the presence of different ground topologies. The processes involved include:

- highly accurate long-term underwater navigation
- autonomous obstacle recognition and avoidance
- autonomous operation monitoring system (system diagnostics)
- reactive mission management system
- case-sensitive track control
- situation-adaptive vehicle controller
- global and local communication and data management

Within Deepc Control Architecture capabilities, the monitoring system is of particular significance. It divides into mission monitoring and health monitoring. The mission monitoring feature is responsible for the mission sequence. It is used to analyse running missions and gives recommendations for any necessary replanning to mission control.

UAH Vehicle Platform and Robot Piloting Module. In the context of the ADVOCATE II project, UAH deploys a ground vehicle that works in a combination of autonomous and teleoperated mode. The vehicle is intended to perform surveillance tasks after hours in a large building composed of corridors, halls, offices, laboratories, etc. For this purpose, UAH is currently deploying the BART (Basic Agent for Robotic Tasks) robot, depicted in figure 4. The operator is in charge of global vehicle navigation by remotely commanding its actuators according to the images that are continuously transmitted through a wireless ethernet link from the vehicle to the base station. Information concerning proximity sensors (the vehicle is equipped with a ring of ultrasound sensors) is also transmitted for monitoring.

A key variable for energy monitoring and diagnosis is the estimated State of Charge (SOC) of the vehicle battery. SOC estimations are based on battery voltage measurements [6]. In order to provide stable SOC estimations, a kalman

filter is used so as to remove gaussian noise in voltage measurements, according to the simplified battery model provided in [3] [4] [5], described by equation 1.

$$\begin{bmatrix} V_{c_k} \\ R_k \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_{c_{k-1}} \\ R_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{-T_s}{C} \\ \frac{1}{K_r} \end{bmatrix} I_{bat_{k-1}} + \begin{bmatrix} w_{1_{k-1}} \\ w_{2_{k-1}} \end{bmatrix} \quad (1)$$

$$V_{bat} = \begin{bmatrix} 1 & -I_{bat_k} \end{bmatrix} \begin{bmatrix} V_{c_k} \\ R_k \end{bmatrix} + v_k$$

where I_{bat} and V_{bat} represent the current and voltage of the battery respectively, R stands for the internal battery resistance, C is the battery capacity, w_1 and w_2 represent the components of the process noise, v stands for the measurement noise, and K_r is a battery dependant parameter that describes the relation between current and voltage. After that, the SOC is computed using the model proposed in [6], as shown in equation 2.

$$\begin{aligned} V_{avg} &= \frac{V_{c_{k-1}} - V_{c_k}}{2} \\ R_{avg} &= \frac{R_{k-1} - R_k}{2} \\ V &= V_{avg} - I \cdot R_{avg} \\ r &= \left(\frac{V_{avg}}{2 \cdot R_{avg}} \right)^2 - \frac{P_{bat}}{R_{avg}} \\ I &= \frac{V_{avg}}{2 \cdot R_{avg}} - \sqrt{r} \\ SOC_k &= SOC_{k-1} - \frac{P \cdot \Delta t}{3600 \cdot C \cdot V} \end{aligned} \quad (2)$$

where P_{bat} is the rated battery power, P denotes the instant power, and Δt represents the time interval in the estimation process.

1.2 Decision Module (DM)

The Decision Module can be considered as the central unit of the ADVOCATE system architecture. The Decision Module needs to have a good knowledge about the whole system. That is why it collects information about all system modules during initialization. The decision module has a generic part and a specific part containing knowledge for decision making, according to diagnosis results. The Decision Module manages the overall diagnosis and recovery process, including control of the monitoring/diagnosis/recovery process, validation of diagnosis and recovery actions (if needed), interaction with human operators (if any) with regards to diagnosis and recovery, integration of uncertainty information provided by the intelligent modules, and conversion of the recovery actions into recovery plans, and subsequent processing of the recovery plans. It is made up of four functional blocks:

- Diagnosis Process Manager. This subsystem is in charge of managing all the interactions related to diagnoses and recovery, from a procedural point of view.
- Initialization Module. This subsystem gathers all actions that take place during initialization. This module is in charge of one action, which does not take place during initialization (answer life checking). This action is related to registration.

- Measures Integrator (Decisions Merger). As there could be several diagnoses or recovery actions attending a single request, once these different results get to the decision module they have to be integrated.
- Recovery Action to Recovery Plan Translator. It is necessary to translate the recovery action into a series of manoeuvres (a recovery plan). The task of this subsystem is to translate a recovery action (a result generated by the intelligent modules) into a recovery plan that can be executed by the piloting module. This module is internal acting upon request of the Diagnosis Process Manager.

The core of the module from a procedural point of view is the Diagnosis Process Manager. This block reacts once an alarm is received or once the operator launches a diagnosis, a monitoring, or a recovery process. From that point, it controls the whole process, by determining when, how, and what module (Intelligent Module, Robot Piloting Module, or MMI) to ask for information, and continuing the process until the final recovery plan is provided to the Vehicle Piloting Module.

1.3 Intelligent Modules

Several Intelligent Modules for each End-User application are currently being developed, using different Artificial Intelligent techniques devoted to solve real problems on operational robots by making use of specific knowledge on them. Intelligent Modules include functionalities providing a diagnosis (identification of sub-system state), a proposed recovery action, or both. The main role of the intelligent modules is to serve as efficient modules for solving diagnosis and recovery action problems. Each intelligent model plugged onto the ADVOCATE II system architecture consists of a generic part and a problem specific part. The problem specific part will typically consist of the knowledge base (neural network, rule base, or Bayesian belief network) to be used to solve the diagnosis and recovery action problem and possibly “bookkeeping” functionality such as data pre- and post-processing. The generic part is the interface and the functionality common to all intelligent modules. An intelligent module may have two different modes of operation such as an off-line mode and an on-line mode. In on-line mode an intelligent module uses its knowledge base to process requests to produce diagnoses and recovery actions. In off-line mode the knowledge base to be used in the on-line mode is constructed, updated, or revised by learning, for instance. The functionality of the off-line mode will depend on the particular technique used in the intelligent module whereas the functionality of the on-line mode is defined by the interface of the generic intelligent module. The present implementation comprises modules based on:

- Bayesian Belief Networks (BBN).
- Fuzzy Logic (FL).
- Neuro-Symbolic Systems (NSS).

1.4 Directory Module

The Directory Module is a central point of the architecture. Developed in Java, it implements part of the UDDI registry. It registers dynamically all the modules plugged on the architecture, provides on request relevant module URLs, checks regularly if the registered modules are still operational, and in case some of them is not any more, informs the others. All the inter-modules communication is based on the SOAP standard, because of its flexibility and lightness. However, in order to support the soft real time constraints of such technical application a complete middleware has been developed including implementation of SOAP extensions to manage timeouts and priorities. This middleware provides the developers of the different modules all the “communication material” for an easy implementation of the communication interfaces. In addition, an Integration Tool and a Test Tool have been developed to provide support also during integration and test phases.

1.5 Configuration Tool

This is an offline, user-friendly application, which eases the production of the XML File describing all the communication interfaces to be developed and that constitutes an input for the SOAP middleware, as well as the XML Configuration File for every modules of the ADVOCATE II system, including the different necessary parameters. By generating a graphical view of the system the user will be able to check the concordance of the configuration files, and to foresee the behaviour of the modules in the system. All this set of tools has been designed to reduce at maximum and make easier the work to adapt ADVOCATE to a new or existing system.

2 Applications

Three end-users are involved in the ADVOCATE II project:

- IFREMER (France) designing Autonomous Underwater Vehicles (AUVs) for scientific applications.
- STN ATLAS Elektronik (Germany) designing Autonomous Underwater Vehicles (AUVs) and semi-AUVs for industrial applications.
- UAH (Spain) designing Piloting Modules for either Autonomous or Remotely Operated Ground Vehicles (AGVs or ROGVs, respectively) for surveillance applications.

2.1 IFREMER Application

Within the ADVOCATE II project, VORTEX will be considered as a working AUV, i.e. without human supervision during a mission execution. Thus, it must be demonstrated that the intelligent diagnostic architecture to be developed in the project can handle automatic problem resolution, without any guidance by

an operator during a mission execution. The different Diagnosis Problems that need to be solved with ADVOCATE are the following:

- Thruster malfunctioning diagnosis
- Energy consumption monitoring

Diagnosis Problem: Thruster Malfunctioning. The VORTEX vehicle is controlled thanks to several thrusters. The controller commands *rpm* can help to perform some simple diagnosis. However, in some situations (damage to the propeller for instance) the vehicle controller will not detect the failure. In the case of thruster failure, the vehicle is no longer controllable as planned. Recovery actions have then to be applied in order to save the vehicle or to prevent the damaging of the environment. The diagnosis may have to be performed by doing specific test manoeuvres in order to get more information on the vehicle behaviour. As a function of the evaluated diagnosis, adaptation of the vehicle controller or the mission objectives may be envisaged. The following data are steadily monitored :

- Current consumption of all thrusters
- commanded *rpm*
- global navigation data of the vehicle

Diagnosis Problem: Energy consumption monitoring. The energy consumption monitoring is mandatory for an AUV since it will directly influence the correct execution of the mission and the safety of the vehicle. Any abnormal energy consumption should be detected and reported. The corresponding recovery actions in this situation will be to determine if the programmed mission can be continued and terminated (maybe by simply reducing the speed) or if some mission plan modification is requested. Data available for assessment are:

- the initial amount of energy
- the current consumption of energy

2.2 STN-ATLAS Application

Concerning DeepC AUV, three main diagnosis problems are stated:

- Assessment of Global Vehicle Behaviour (Motion Analysis)
- Propulsion and Thruster Diagnostics
- Sensor Malfunction

Diagnosis Problem: Global Vehicle Behaviour. An Autonomous Underwater Vehicle has to perform a long endurance mission about 60 hours or more. During that time it may be possible that the behaviour of the vehicle is not as it is expected in respect of the control input. For instance it shows a tendency to turn left although all controls are in a neutral position. Because of the fact

that DeepC is fitted with an adaptive controller, it may be more likely that a control input necessary for a wanted behaviour is uncommon. The first aim of Advocate will be a monitoring and assessment of the motion characteristic and the control inputs. After the analysis of the measured behaviour an appropriate Recovery Action has to be proposed. The following input Data can be provided by the AUV system:

- Thruster Diagnosis
- Navigation Data: speed vector (North, East, Down), position (Latitude, Longitude, depth)
- Controller Data: flap angle, thruster *rpm*

The behaviour of the vehicle can be assessed by comparison with a numerical model or by a knowledge-based description or both. Using a model for comparison a high quality model must be implemented and it takes a huge amount of processing. A rule- or expert-knowledge based comparison is faster, maybe more robust, but does not comprise all vehicle states. Further it will be difficult to validate over the complete envelope of motion capability. But because DeepC will run 90 % of its mission time straight ahead it may be sufficient to regard this vehicle state. In this case the knowledge-based description is preferred.

Diagnosis Problem: Thruster and/or Actuator Malfunction. Propulsion and steering of the DeepC AUV is achieved by propellers (thrusters) and rudders (flaps). As mechanical devices they are prone to getting entangled with floating obstacles. In worst case parts may dismantle due to faulty installation or external forces. The vehicle controller will not detect a possible failure. In a possible failure case the vehicle is no longer controllable. It is absolutely necessary to perform recovery actions to save the vehicle or to prevent the damaging of the environment. Failures may vary in their significance: with increasing amount of foreign substances between (or clamped to) moving parts the more additional torque on the actuators will be generated. The following data are steadily monitored at a time interval dt or on request of the Motion Analysis Diagnosis or on request of a check module:

- Current consumption of all motors
- commanded *rpm* or angle value
- status of controller

From this data the health status of the thruster/actuators are generated by comparing current values with ideal values. The following Diagnosis is expected:

- Actuator ok, entangled, or lost
- Motor/controller damaged

Diagnosis Problem: Sensor Malfunction. As a fully autonomous system, DeepC has to rely on its sensors to survive operational. Among the most important sensors are:

- the Inertial Measurement Unit to measure the accelerations of the vehicle
- the CTD Probe to measure conductivity, temperature and depth
- the obstacle avoidance Sonar

It is possible that failures occur so that the values, the sensors currently reporting, are corrupted or too noisy. A special problem of the sonar is, that possible obstacles may be hidden in a cluttering environment. This is probably true for an operation near the seafloor. In this case images cannot be analysed automatically because the results of the image processing algorithms are poor. There is a risk of damaging the vehicle by an unforeseen collision. It can be considered to have a poor image whenever:

- The distance to the ground/surface is small so that a high portion of bottom reverberation is present, or
- High Seastate or other broad band noise or sound sources (e. g. ship noise) are in the vicinity of the sonar receiver

2.3 UAH Application

During the execution of the mission, several problems or failures can occur yielding the vehicle to abnormal behaviour with respect to the expected one, and thus, impeding proper finalisation. ADVOCATE II will be used to do diagnosis and/or recovery actions for those failures, using the modular and intelligent diagnosis systems developed in this project.

Diagnosis Problem: Energy problem. The vehicle electrical energy can run out during operation impeding the complete finalisation of the mission. That could become a serious disadvantage particularly if the vehicle is operating far away from the base station, or if several vehicles are simultaneously being remotely monitored by just one operator. From continuous monitoring of the vehicle power consumption along the planned track and prediction of the remaining power, the likelihood of mission success can be forecasted. In case the predicted power consumption exceeds the available battery capacity (plus some reserve), the mission conditions should be accordingly replanned. Then, the battery State Of Charge (SOC, hereinafter) requires continuous monitoring so as to envisage the most appropriate action to take, according to the remaining vehicle energy. For making the decision upon Energy problems, the Intelligent Module (based on Bayesian Belief Networks) in charge of that task will be supplied with the following Basic Information.

- Initial SOC (percentage over full nominal charge)
- Planned mission and prediction of power requirements
- Speed profile

Diagnosis Problem: Actuator malfunction. If at any time during the mission some failure in the drive actuators arises (malfunction of motors, soft obstacles in the axes, sliperages, etc), mission execution and proper finalisation will not be possible any longer, or at least, vehicle safety and stability would be compromised. An appropriate diagnosis for correct detection of this type of faults is mandatory in order to start some recovery action to compensate for the vehicle failure, or to proceed to execute additional test manoeuvres so as to gather as much information as possible to get further diagnosis. Whenever the vehicle piloting module detects some actuator problems it sends an alarm to the decision module which is in charge of providing appropriate diagnosis and/or recovery actions. For making the decision upon actuator failures, the Intelligent Module (based on Neuro-Symbolic Systems) in charge of that task will be supplied with the following Basic Information.

- Vehicle Dynamic Model
- Commanded actuation
- Current vehicle state

Diagnosis Problem: Sensor related motion problems. Ultrasound sensorial information is the main input environment data used for navigation and thus it works as a safety system for collision avoiding in either the autonomous and the teleoperated modes. Nonetheless, there are some obstacles (short height obstacles, indeed) that can not be detected by ultrasound means. In case the AGV finds an obstacle of this type along its way, the vehicle will inevitably collide with the obstacle, which can even be dragged. In order to recover from these situations, the vehicle piloting module will carry out a first coarse detection of the problem intended to provide an alarm to the Decision Module. Upon the alarm, the Decision Module will require diagnosis on the problem to the appropriate Intelligent Module (based on Fuzzy Logic). If the diagnosis provided by the FL IM confirms the occurrence of the problem, a recovery action will be generated so as to get the vehicle rid of the obstacle and resume the mission afterwards. The Vehicle Piloting Module provides the following information concerning real and measured data:

- Battery voltage and consumption
- Commanded and angular velocity of left wheels
- Commanded and measured angular velocity of right wheels
- Commanded and measured linear velocity
- Ultrasound range measures

3 Conclusions

The main objective of the ADVOCATE II project is to develop a software architecture to allow the implementation of intelligent control modules for underwater and ground robotic applications, as described in this paper, in order to increase their reliability. The interest of such a concept from the marketing point of

view has been demonstrated by a market study. Additional ongoing information concerning the ADVOCATE II project can be found at the project web site: <http://www.advocate-2.com> .

Acknowledgments. This work is supported by the European Commission (IST-2001-34508).

References

1. Advocate Consortium: ADVOCATE: ADVanced On-board diagnosis and Control of Autonomous sysTEms. IPMU 2002, Information Processing and Management of Uncertainty, Annecy, France, July 1–5, 2002.
2. Advocate Consortium: ADVOCATE II Technical Annex: Description of Work. January 2002.
3. M. Hemmingsson: A Powerflow Control Strategy to Minimize Energy Losses in Hybrid Electric Vehicles. Department of Industrial Electrical Engineering and Automation. Lund Institute of Technology. Lund University. ISBN: 91-88934-11-X. Printed in Sweden. 1999.
4. E. Karden, P. Mauracher, and Friedhelm Schope. Electrochemical modelling of lead/acid batteries under operating conditions of electric vehicles. *Journal of Power Sources*. Elsevier, **64** (1997) 175–180.
5. P. Mauracher, and E. Karden. Dynamic modelling of lead/acid batteries using impedance spectroscopy for parameter identification. *Journal of Power Sources*. Elsevier, **67** (1997) 69–84.
6. S. Reehorst. Battery State-of-charge Calculations. Power and Propulsion Office. Appendix B1. NASA Glenn Research Center. 2002.

Segmentation of Traffic Images for Automatic Car Driving

Miguel Ángel Patricio¹ and Darío Maravall²

¹ Departamento de Informática
Universidad Carlos III de Madrid
`mpatrici@inf.uc3m.es`

² Department of Artificial Intelligence
Faculty of Computer Science
Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Madrid, Spain
`dmaravall@dia.fi.upm.es`

Abstract. This paper addresses the automatic analysis and segmentation of real-life traffic images aimed at providing the necessary and sufficient information for automatic car driving. The paper focuses on the basic task of segmenting the lane boundaries. As the general objective is to build a very robust segmentation module, able to cope with any kind of road and motorway and for any kind of surroundings and background, either rural or urban, we face a complex problem of texture analysis and classification which we have approached by applying the frequency histogram of connected elements (FHCE). To assure an efficient design of the segmentation module, a thorough experimentation with numerous traffic images has been undertaken. In particular, the optimum design of the crucial parameters of the FHCE (namely, the structurant morphological element, the connectivity level and the scanning window) has been carried out with special care. Experimental results are finally presented and discussed.

1 Introduction

During the last two decades there has been increasing interest and activity in the development of advanced transportation systems for local roads and motorway, as well as for urban traffic. Vision-based driver assistance is one of the most promising and challenging aspects of the effort aimed at designing and building automated aids for the improvement in traffic infrastructures and for safer transport systems [1]–[3].

One of the main goals of a vision-based traffic system is to segment the road surface and to detect any vehicle on the road. Although the identification of the road surface boundaries and the detection of the relevant elements which are present within the road limits is quite trivial for a human driver, accurate and reliable automatic segmentation of the road and the existing elements is still a challenging task.



Fig. 1. Two instances of traffic images: (a) well-structured road and (b) rural road.

In this paper, we address the analysis and segmentation of traffic images aimed at providing the necessary and sufficient information for driver assistance and, ultimately, for automatic car driving at low speeds and under complete human operator remote control. More specifically, the paper focuses on the basic task of segmenting the road boundaries. As the final objective is to build a very robust segmentation module, able to cope with any kind of road or motorway and with any type of surroundings and road backgrounds, we are dealing with a rather complex problem of texture analysis and classification.

Figure 1 shows two typical traffic images: (a) corresponds to a well-structured road and (b) represents a rural road in which the road boundaries are much harder to discriminate. The three most popular methods for detecting lane and road boundaries like those appearing in Figure 1 are histogram-based segmentation [4]–[6]; model-based lane tracking [7]–[9]; and pattern recognition techniques based on texture and color discriminant features [10] and [11].

Histogram-based segmentation for road detection has the advantage of simplicity and real-time performance, although its accuracy is sometimes rather low, in particular for highly varying road boundaries and backgrounds. On the contrary, the techniques based on texture and color discriminant features are, in general, much more efficient, but at the cost of an intensive design process and computational complexity. In this paper, we have taken an intermediate approach, as we have applied a novel computer vision technique that combines the simplicity of the conventional histogram and the discriminant power of texture features.

Roughly speaking, textures or spatial structures can be divided into two groups: (1) regular structures and (2) irregular structures. The first group is typical of artificial or man-made environments and objects, whereas the second group corresponds to the natural world. According to this distinction, the techniques for texture analysis and recognition can be classed as (1) syntactic methods and (2) statistical methods, respectively; although the latter can also be applied to regular structures. This division of texture analysis and recognition methods is equivalent to the well-known distinction in the discipline of pattern recognition between syntactic and statistical or metric methods. Most of the textures appearing in traffic images are irregular by their very nature, so that statistical methods are compulsory in this application domain. The statistical methods for texture analysis and recognition can be divided into two main groups: (1) methods based on the unidimensional histogram and (2) methods based on bidimensional histograms, in which the idea of dimension is not associated with the discriminant variable or attribute, but with the notion of space. Thus, the unidimensional histogram is obtained from the values taken by the particular discriminant variable under consideration at each individual pixel, which is, obviously, a spatial predicate. Similarly, the bidimensional histogram is obtained from the values taken by the same specific attribute at each pair of pixels, having some previously defined spatial restriction or predicate. In other words, what differentiates unidimensional histograms from bidimensional histograms is not the dimension of the space of the possible discriminant variables -which is actually always unidimensional-, but the dimension of the respective spatial predicate. Therefore, the key point is the physical space of the pixels, rather than the space of the discriminant attributes. Bidimensional histograms, computed from co-occurrence matrices, provide useful information about the spatial distribution -about a specific two-point spatial relationship in actual fact- of the particular discriminant variable at hand, which is obviously relevant information as far as texture analysis and recognition are concerned. However, they have two serious drawbacks: their excessive computational load and, in particular, the curse of dimensionality that produces a considerable amount of irrelevant information and noise. On the contrary, unidimensional histograms, which are computationally speaking very attractive and even discriminant enough to efficiently solve the segmentation problem in many applications, do not provide as much discriminant information, in particular spatial and textural information, as bidimensional histograms do.

The authors have recently introduced a novel concept which intends to capture the strengths of both histograms [12] and [13]. This concept, which we have called the frequency histogram of connected elements (FHCE), is a generalization of the unidimensional histogram and, at the same time, is related to the co-occurrence matrix and, hence, to the bidimensional histogram as well. Being a conventional unidimensional histogram, the FHCE incorporates all the computational advantages, in terms of both simplicity and speed, inherent to histogram-based segmentation methods. Simultaneously, it includes information about the spatial distribution of the specific discriminant feature in the digital image, as bidimensional histograms do. The FHCE concept has an additional advantage in comparison to bidimensional histograms, as it is based on a much more powerful spatial function than the simple two-point relationships, typical of bidimensional histograms and co-occurrence matrices. This is the concept of structuring element or spatial predicate, which is somewhat, but not entirely, related to the structuring element concept used in morphological image processing [14]. Furthermore, there is yet another interesting advantage of the FHCE, as compared with the conventional unidimensional histogram. This is its flexibility with regard to the range of values of the discriminant variable, which is absolutely rigid in conventional unidimensional histograms. This means that we can juggle with an interesting degree of freedom, which we have called the connectivity level, for texture analysis and recognition. The main contribution of the paper is the application of this novel concept to the segmentation of textured images taken from real-life traffic scenes. To this end, we have carried out a thorough experimentation with the two basic parameters of the FHCE, namely, the *structural element* and the *connectivity level* in order to attain the desired efficiency and robustness in the segmentation module. Another crucial parameter is the scanning window, or region of interest, that determines the spatial domain for texture analysis with the FHCE. A very extensive set of traffic images are used to assure an efficient design of the segmentation module. Experimental results are finally presented and discussed.

2 Traffic Lanes Segmentation

2.1 The Frequency Histogram of Connected Elements

Let us proceed now with a brief description of the theoretical foundation of the FHCE. But before doing so, we would like to remark that although this novel concept can be applied to any type of discriminant feature, either sensory or abstract, for the sake of clarity, we shall focus only on the typical grayscale intensity.

The Neighborhood Concept. Let $\{I(i, j)\}_{N \times M}$ be a digital image. If we denote by (i, j) the coordinates of a generic pixel, the neighborhood of this pixel is defined as follows:

Let $\{I(i, j)\}_{NxM}$ be a digital image. If we denote the coordinates of a generic pixel as (i, j) , the neighborhood of this pixel, $\mathcal{N}$, is defined as follows:

$$\begin{aligned}\mathcal{N} &\triangleq \{\varphi_{i,j} \subset \{I(i, j)\}_{NxM}\} \\ \varphi_{i,j} &= \{\forall(k, l) / D[(k, l), (i, j)] \text{ is true}\}\end{aligned}\quad (1)$$

where D is a predicate defined by a distance-based condition. For instance, a valid definition of the neighborhood of a pixel $I(i, j)$ can be given by the set:

$$\varphi_{i,j}^{r,s} = \{\forall(k, l) / \|k - i\| \leq r \text{ and } \|l - j\| \leq s\} ; r, s \in \mathbb{N} \quad (2)$$

which indicates that the neighborhood of the pixel is formed by a set of pixels whose distances are not greater than two integer values r and s , respectively.

The Connected Element Concept. We define a connected element as follows

$$C_{i,j}(T) \triangleq \varphi_{i,j}^{r,s} / I(k, l) \in [T - \varepsilon, T + \varepsilon] , \forall(k, l) \in \varphi_{i,j}^{r,s} \quad (3)$$

where I is the grayscale intensity or brightness of pixel (k, l) . In other words, a connected element is any neighborhood unit such that its pixels have a grayscale level close to a given grayscale level T .

The Frequency Histogram of Connected Elements. The frequency histogram of connected elements (FHCE) is defined as:

$$H(T) = \sum_{\forall(i,j) \in \{I\}} C_{i,j}(T) \quad (4)$$

$$0 \leq T \leq I_{max} - 1$$

That is to say, $H(T)$ approximates a density function for a random event occurring in a digital image $\{I(i, j)\}_{NxM}$. This event is related to the idea of connected element, which in turn is related to the pseudo-random structure of the grayscale intensity distribution of a particular texture. From an intuitive standpoint, we can understand the information conveyed by the function $H(T)$: high values of $H(T)$ indicate that there is a sizeable number of connected elements in the image whose grayscale levels are homogeneous and close to level T .

Obviously there is no universal connected element valid for any domain application. In the design leading to the FHCE there is a critical and domain-dependent step, which is responsible for the selection of the parameters defining the optimum, connected element. Such parameters are: (1) the morphological component and (2) the connectivity level.

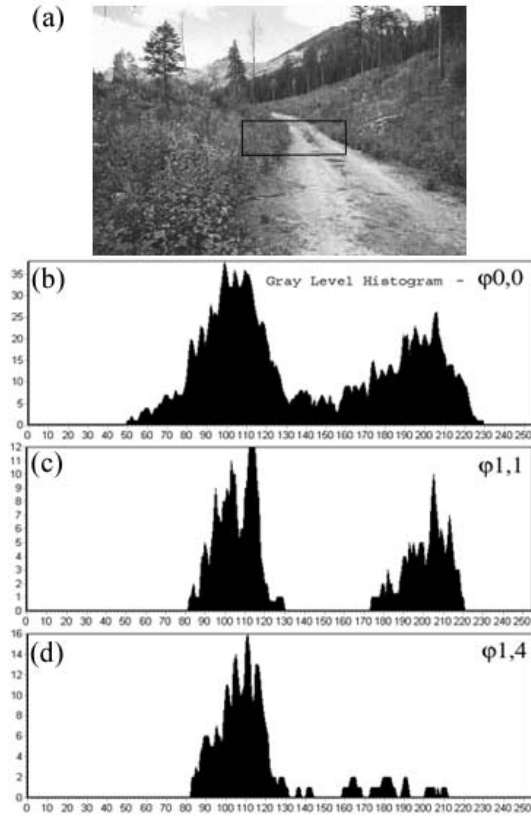


Fig. 2. Portion of a thin crack in wood and several FHCEs obtained using different morphological components.

2.2 Morphological Components

We mean by morphological components those that determine the shape of the connected element; i.e., what we have called neighborhood. Obviously, these parameters are very dependent on the application domain, and final system efficiency will rely on the correct selection of the morphological components.

The selection of the suitable morphological components for a particular application is not an easy task, and it requires a thorough empirical analysis. Even if specialized algorithms are used to choose the optimum set of morphological components, the final goal of the computer vision application at hand will determine the results. Thus, any knowledge we have about the application domain –i.e. our a priori knowledge of the digital images to be automatically analyzed– is crucial for the correct selection of the morphological components. As an illustration of this fact, just take a look at Figure 2, where different FHCEs have been computed and compared. These FHCEs have been obtained using different morphological components.

From the FHCE shapes, it is apparent that the correct choice of the morphological components is vital for the final success of the problem of detecting the road boundaries. In Figure 2a we have selected a region of interest -denoted by a square- in order to illustrate the methodology for designing the morphological parameters. For $r = 0$ and $s = 0$ (Figure 2b) the pixel's neighborhood is only composed by itself and, thus, every pixel of the region hold the connected element restriction given by expression (2). As a conclusion of this fact, the conventional grayscale level histogram is a particular case of the FHCE. In the same figure are depicted the FHCE corresponding to another possible morphological parameters and clearly the case of Figure 2c ($r = 1$; $s = 1$) is the optimal choice, as it perfectly discriminates the connected elements belonging to the road -the distribution on the left side of the FHCE- from those belonging to the boundaries. On the contrary, the morphological component of the Figure 2d ($r = 1$; $s = 4$) do not provide any information at all about such discrimination.

2.3 Connectivity Level

The connectivity level of a FHCE depends on the value ε that appears in the definition of connected element given by equation (3) and determines the restrictions that every pixel in a neighborhood must meet in order to form a connected element. As is the case for morphological components, the connectivity level is highly dependent on the application domain, and its correct choice is a matter of thorough experimentation. To illustrate this, let us compute several FHCEs with different connectivity levels. As before, we shall use an image of a rural road -see Figure 3-.

The connectivity level increases from null connectivity in Figure 3b to highest connectivity in Figure 3d. Null connectivity means that every neighborhood defined by a morphological component is considered as a connected element, which in fact leads to the conventional grayscale level histogram. Looking at the different FHCEs in Figure 3 is clear that the choice of an optimum connectivity level is of vital importance. Thus, a clear bimodal distribution appears in Figure 3c, in which the first distribution -on the left-hand side- represents the connected elements formed by the road and the other distribution corresponds to the limits.

2.4 Implementation Details

Through experimentation, we have found that the best results are obtained when applying a small window over the whole image. As happens with any other local operator, the key issue is to correctly select the size of the scanning window. For texture analysis, the basic idea is to apply a window whose size is big enough to capture the essential structure of any texture present in the image.

Another important point is the scanning process, i.e. the way the window is passed over the image. The common scanning process is to divide the whole image into a set of non-overlapping, adjacent cells or windows. This solution is very attractive from the computational point of view, but very inefficient for capturing the real textures occurring in the image. Thus, it is advisable to

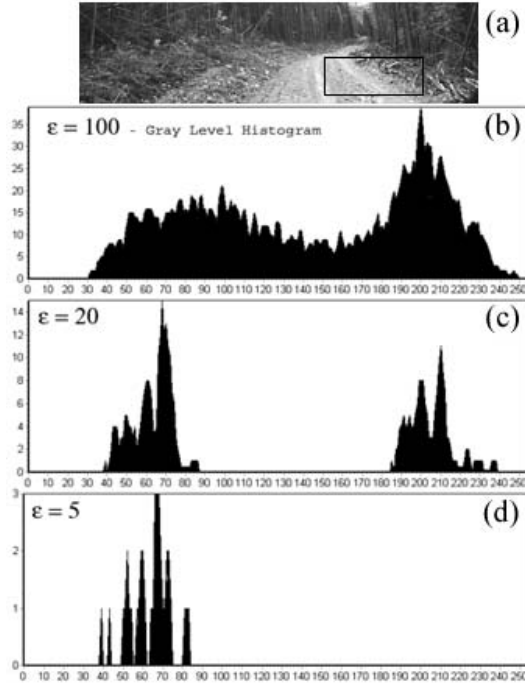


Fig. 3. Rural road image and several FHCEs obtained with different levels of connectivity, increasing from (b) to (d).

explore the image in more depth, where the extreme option is to apply the scanning window to every pixel in the image. This is extremely cumbersome for computational reasons, although optimum for automatic texture analysis. The latter option bears a tremendous computational load, as the FHCE must be computed for each pixel as many times as there are possible values of T , while, at the same time, checking whether the neighborhood of the current pixel meets the condition of connected element. Remember that for a typical grayscale level image there are 256 different values of T .

In order to deal with real-time applications, we have implemented an *ad hoc* procedure for obtaining the FHCE, which is based on computing, for each pixel, the maximum and the minimum grayscale level values in its neighborhood. The pixel neighborhood is classified as a connected element if the difference between the maximum and the minimum values divided by the dynamic range of the grayscale level histogram of the window centered on the current pixel is small. For instance:

$$\frac{i_{max} - i_{min}}{I_{max} - I_{min}} < \alpha \quad (5)$$

or

$$\left(1 - e^{-\beta |i_{max} - i_{min}|}\right) < \alpha \quad (6)$$

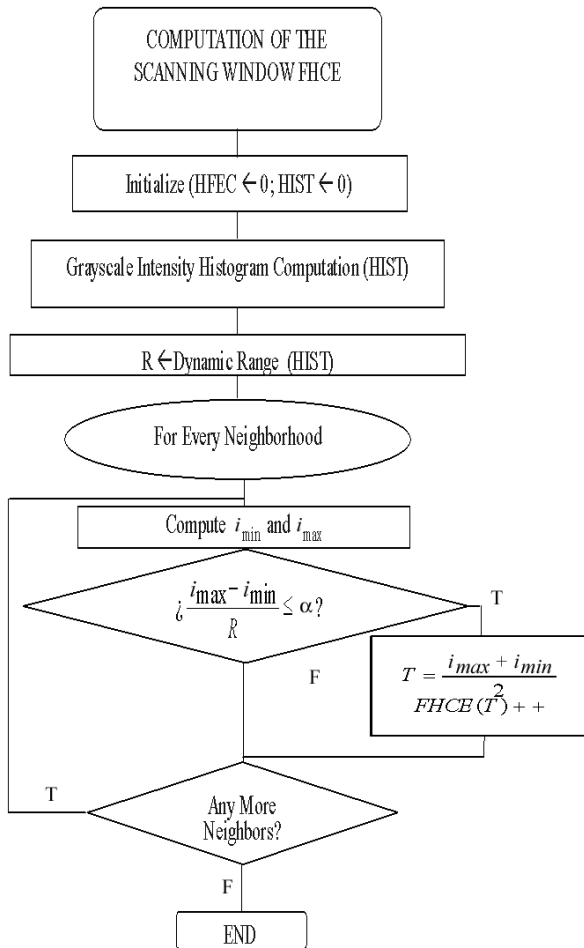


Fig. 4. Flowchart of the FHCE computation

where

$$\frac{2}{I_{max} - I_{min}} \leq \beta \leq \frac{5}{I_{max} - I_{min}} \quad (7)$$

in which i_{max} and i_{min} are the maximum and minimum grayscale levels in the neighborhood, respectively, I_{max} and I_{min} are the maximum and the minimum brightness levels in the window, and α is a very small numerical value – for instance, $\alpha = 0.1$ -. If conditions (5) or (6) hold, then the respective neighborhood is labeled as a connected element, and a new event is added to the FHCE for $T = (i_{max} + i_{min})/2$. More sophisticated estimators of T may be attempted –for instance, the mean or the median grayscale levels of the neighborhood-, but there is a considerable additional computational load and only a slight improvement in the discrimination task. Figure 4 shows the flowchart of the FHCE computation.

3 Experimental Results

As for any other contextual segmentation procedure aimed at exploiting the local or spatial information of the image under analysis, the key issue when applying the FHCE concept is a correct selection of the scanning window's size. In our particular application of segmenting traffic lanes, we have found that a window of 80x30 pixels seems to be optimum in most cases. After exhaustive experimentation with numerous digital images of rural roads we have selected a 3x3 window as the neighborhood or morphological structure. To conclude the selection of the FHCE parameters, the connectivity level that a particular neighborhood should possess to be considered as such must be selected. The FHCE is computed for each image portion by shifting through all the pixels a window of the same 3x3 shape than the neighborhood. This scanning process is performed by means of a top-bottom and left-right movement and by computing at each pixel the maximum and the minimum gray level within its neighborhood. Each pixel's neighborhood is classified as a connected element if and only if the difference between the maximum and the minimum values is small as compared with the dynamic range of the histogram in the whole window. After experimental work we have chosen a 20% ratio, which is a good compromise between road areas and their limits. Therefore, for a neighborhood to be labeled as connected element the following condition has to be checked: $((i_{max} - i_{min}) / (I_{max} - I_{min})) \leq 0.2$ (see Equation (5)). Thus, if a particular neighborhood possesses a gray-level variability less than twenty percent of the dynamical range of the global window the corresponding pixel is a connected element and the FHCE will compute a new event with value $T = (i_{min} + i_{max}) / 2$.

As commented above, we have experimented with many digital images of roads and highways in order to select the most efficient values of the FHCE parameters –i.e., the morphological component, the connectivity level and the scanning window. Once these parameters were settled with the values reported in the previous paragraphs, a comprehensive experimentation with numerous images was carried out to test the FHCE performance in segmenting the lane boundaries. We have obtained excellent results for all type of rural roads, local roads and highways, as the FHCE is able to perfectly discriminate the corresponding lane boundaries. To illustrate the FHCE performance in segmenting lane boundaries, Figure 5 shows the segmentation results obtained for five different images, ranging from rural roads to highways.

4 Conclusions

The main contribution of this paper is the application of the frequency histogram of connected elements (FHCE) to the segmentation of textured images taken from transportation infrastructures: more specifically, to the segmentation of lane boundaries. To this end, we have carried out a comprehensive experimentation with the two basic parameters of the FHCE, namely, the morphological component or structuring element and the connectivity level, in order to attain

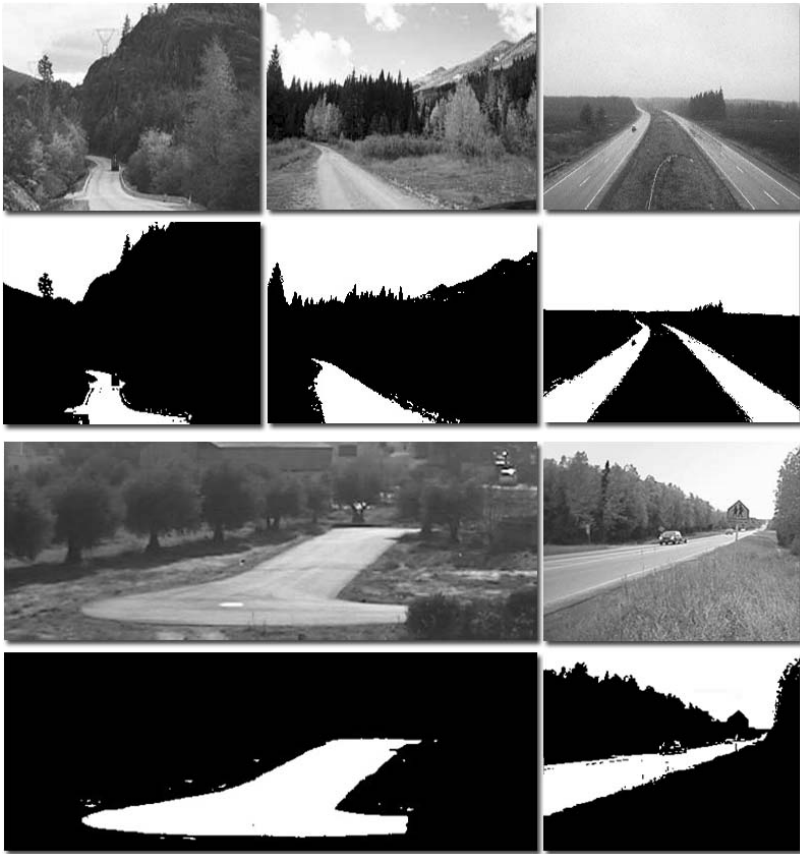


Fig. 5. Results of the segmentation process by applying the FHCE concept to several roads images. In all the cases, notice the efficiency in segmenting the lanes appearing in the images.

the desired efficiency and robustness in the segmentation task. Another crucial parameter in our experimentation has been the scanning window, or region of interest for texture discrimination, that determines the spatial domain for the texture analysis with the FHCE. A very extensive repertoire of real-life traffic images has been used for the parameters selection. After settling the FHCE parameters, an exhaustive test with numerous traffic images, ranging from rural roads to highways, has been carried out. The experimental results have demonstrated that the FHCE can be considered an excellent instrument for the analysis and segmentation of road boundaries.

Acknowledgements. This work has been partially supported by the Spanish Ministry of Science and Technology, project DPI2002-04064-CO5-05.

References

1. Vlacic L., Parent M., and Harashima F.: *Intelligent Vehicle Technologies*. Butterworth Heinemann, Oxford, (2001).
2. Handmann, U., Kalinke, T., Tzomakes, C., Werner, M., Seelen, W.V., "An image processing system for driver assistance", *Image and Vision Computing* 18, 2000, 367–376.
3. Bertozzi, M., Broggi, A. Cellario, M., Fascioli, A., Lombardi, P., and Porta, M., "Artificial Vision in Road Vehicles", *Proceedings of the IEEE*, Vol. 90, No. 7, July 2002, 1258–1271.
4. Ran, B. and Liu H.X., "Development of A Vision-Based Real Time Lane Detection and Tracking System for Intelligent Vehicles", 1999.
5. Gonzalez, J. P. and Özgüner, Ü, "Lane Detection Using Histogram-Based Segmentation and Decision Tree", in *Proc. of the IEEE Intelligent Transportation Systems*, 2000, 346–351.
6. Charbonnier, P., Nicolle, P., Guillard, Y., and Charrier, J., "Road boundaries detection using color saturation", in *Proc. 9th Eur. Signal Processing Conf.*, 1998.
7. Goldbeck, J. Graeder, D., Huertgen, B., ErnstS., and Wilms F., "Lane following combining vision and DGPS", in *Proc. IEEE IV*, 1998, 445–450.
8. Kim, K.T., Oh, S.Y., Kim, S.W., Jeong, H., Lee, C.N., Kim, B.S., and Kim, C.S., "An autonomous land vehicle PRV II: Progress and performance enhancement", in *Proc. IEEE IV*, 1995, 264–269.
9. Goldbeck, J. and Huertgen, B., "Lane detection and tracking by video sensors", in *Proc. IEEE Intelligent Transportation Systems*, 1999, 74–79.
10. Schneiderman, H., Nashman, M., A discriminating feature tracker for vision-based autonomous driving. *IEEE Trans. Robotics and Automation*, 10(6), 1994, 769–775.
11. Konishi, S., Yuille, A.L., "Statistical cues for domain specific image segmentation with performance analysis". *Proc. CVPR'2000*, 1125–1132.
12. Patricio, M.A., Maravall, D.: *Wood Texture Analysis by Combining the Connected Elements Histogram and Artificial Neural Networks*. In J. Mira, A. Prieto (Eds). *Bio-inspired Applications of Connectionism*, LNCS 2085, Springer, Berlin, 2001, 160–167.
13. Maravall, D., Patricio, M.A.: *Image Segmentation and Pattern Recognition: A Novel Concept, the Histogram of Connected Elements*. In D. Chen and X. Cheng (Eds). *Pattern Recognition and String Matching*, Kluwer Academic Publishers, (2002).
14. Soille, P.: *Morphological Image Analysis*. Springer, Berlin, (2003).

Vision Based Intelligent System for Autonomous and Assisted Downtown Driving

Miguel Ángel Sotelo, Miguel Ángel García, and Ramón Flores

Department of Electronics, University of Alcalá,
Alcalá de Henares, Madrid, Spain,
{michael,garrido,flore}s@depeca.uah.es
<http://www.depeca.uah.es>

Abstract. Autonomous and assisted driving in city urban areas is a challenging topic that needs to be addressed during the following ten to twenty years. In the current work an attempt in this direction is carried out by using vision-based systems not only for autonomous vehicle driving, but in helping the driver recognize vehicles, and traffic signs. Some well consolidated results have been attained on a private test circuit using a commercial Citroen Berlingo as described in this paper.

1 The Global Concept

The work presented in this paper is in the frame of the ISAAC Project, whose general layout is depicted in figure 1. The ISAAC project undertakes the challenge of multivehicle cooperation in an urban-like environment by deploying three operational prototypes. Each vehicle is equipped with a GPS receiver and a colour vision system, so that either autonomous or manual navigation can be carried out. On the other hand, there is also a module onboard each vehicle which is in charge of providing assistance to the main driving system (either autonomous or manual) for safety enhancement during navigation.

The work described in this paper focuses on the development of an onboard vision based system for autonomous and assisted driving on urban city areas. This implies to solve the problem of intelligent unmanned mission execution using vision as the main sensor. According to this, the following remarkable challenges arise: lane tracking on non-structured roads (roads with no lane markers), sharp turn manoeuvres in intersections (very usual in urban areas), vehicle detection, and traffic sign detection and recognition. The system achieves global navigation by switching between Lane Tracking and Intersection Navigation, while detecting vehicles, and traffic signs in its local surrounding. The detailed description of the algorithms developed for lane tracking and intersection navigation is provided in [14]. Nonetheless, a brief summary of both algorithms is presented in this paper for completeness purposes.

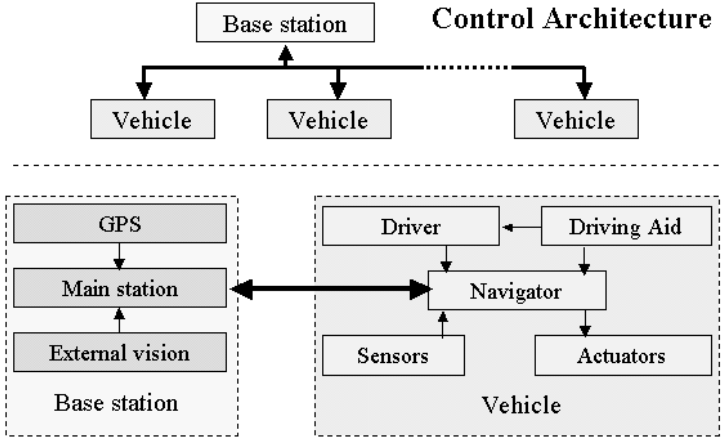


Fig. 1. General layout of the ISAAC Project.

2 Lane Tracking and Intersection Navigation

The mission of this vision based task is to provide correct lane tracking between two consecutive intersections. A polynomial representation is used to model the road edges in the image plane [1] [2]. According to these previous considerations the incoming image is on hardware re-scaled, building a low resolution image of what we call the Area of Interest (AOI), comprising the nearest 20 m ahead of the vehicle. The AOI is segmented basing on colour properties and shape restrictions. Image segmentation must be carried out by exploiting the cylindrical distribution of colour features in the HSI (Hue, Saturation, and Intensity) colour space, bearing in mind that the separation between road and no road colour characteristics is non linear. From the analytical point of view, the difference between two colour vectors in the HSI space can be established by computing the distances both in the chromatic plane, $d_{chromatic}$, and in the luminance axis, $d_{intensity}$, as described in equations 1 and 2.

$$d_{intensity} = |I_p - I_i| \quad (1)$$

$$d_{chromatic} = \sqrt{(S_p)^2 + (S_i)^2 - 2S_pS_i\cos\theta} \quad (2)$$

with

$$\theta = \begin{cases} |H_p - H_i| & \text{if } |H_p - H_i| < 180^\circ \\ 360^\circ - |H_p - H_i| & \text{if } |H_p - H_i| > 180^\circ \end{cases} \quad (3)$$

where H_p , H_i , S_p , S_i , I_p , and I_i represent the Hue, Saturation and Intensity values of the pattern (p) and given (i) pixels under comparison, respectively. As can be readily derived from the previous equations, $d_{chromatic}$ measures the distance

between two 2D colour vectors in the chromatic plane while $d_{intensity}$ provides the luminance difference between the pattern pixel and the pixel under consideration. According to this, a cylindrical surface of separation between the road and non-road classes is proposed in an attempt to decouple chromatic changes from luminance changes, as the latter are much greater in outdoor environments despite intensity is not a determinant characteristic in the colour segmentation process. In other words, any given pixel i will be classified as road if the chromatic distance ($d_{chromatic}$) to the colour pattern is below some threshold T_{chrom} , and the intensity distance ($d_{intensity}$) is lower than some T_{int} . This constraints the road pixels features in a cylinder around the pattern colour vector. On the other hand, to account for road shape restrictions, thresholds are affected by an exponentially decay factor yielding new threshold values that are updated depending on the distance between the current pixel and the previously estimated road model. Once the segmentation is accomplished, a time-spatial filter removes non-consistent objects in the low resolution image, both in space and time (sporadic noise). After that, the maximum horizontal clearance (absence of non-road sections) is determined for each line in the AOI. The measured points are fed into a Least Squares Filter with Exponential Decay [13] that computes the road edges in the image plane as well as the central trajectory of the road using a second order (parabolic) polynomial.

On the other hand, intersection navigation is completely vision based, and accounts for any angular value between the intersection branches. A simple monocular colour vision system is proposed to navigate on intersections (indeed the same system utilised for road tracking). This fact becomes a major issue, from the technological point of view, as cheap prototype navigation systems will be possible. Two basic manoeuvres can be executed at an intersection: on one hand the vehicle can change its moving direction by turning left or right; on the other hand the vehicle can go ahead and cross the intersection by keeping its current direction. The problem of crossing an intersection is basically the same of tracking the lane, and thus the same algorithmic solution is provided for this kind of manoeuvre. On the contrary, turning right or left at an intersection is quite a different problem that needs to be addressed in a different manner. At an intersection, the vehicle will start a turning manoeuvre until the new road to be tracked is perceived with a sufficiently reliable perspective. Vehicle localisation during the turn is reinforced using a Markov stochastic process, in what will be referred to as Markov Localisation Process [15] hereinafter. For this purpose, the angular trajectory described by the vehicle during the turn is modelled by a random variable denoted by ξ . A probability density function is calculated for all possible positions along the localisation space. Such a function is updated at each iteration time under the typical Markov assumptions, and in doing so it becomes a Markov stochastic process. For further details on this algorithm the reader is referred to [14].

3 Vehicle Detection

Vehicle detection is accomplished by using a monocular colour vision system. Using one single image leads to some limitations on the kind of obstacle that can be detected but provides a simple and fast method compared to optical flow based methods [8]. Other vehicles moving in the same or opposite lane can be reliably detected using the road shape and an estimation of the road width to predict the exact area of the image where the obstacles are expected to appear. Vehicles are then characterised by symmetry and edges features, within the estimated road, as far as usual vehicles have quite a distinguishable artificial shape and size that produces remarkable vertical edges in filtered images.

3.1 Searching Area

The execution time is abruptly reduced by limiting the obstacles detection to some predefined area where the obstacles are more likely to appear. This commonly called Region of Interest (ROI) is a rectangular area covering the central part of the image. In order to robustly detect and track vehicles along the road, two consecutive processing stages are necessary. In the first step vehicles are localised basing on the previously mentioned colour and shape properties, while in the second one the already detected vehicle is tracked using a real time estimator. A detailed description of both processes is given below.

3.2 Vehicles Detection

The identification of other vehicles is performed according to vertical edge and colour symmetry characteristics, together with temporal constraints for consistence purposes, under the assumption that vehicles generally have artificial rectangular and symmetrical shapes that make their vertical edges easily recognisable from the rest of the environment. This is quite a realistic situation that can be exploited in practice.

Vertical edge and symmetry discriminating analysis. A first discriminating analysis is carried out on the ROI basing on colour and vertical edge features. It permits to obtain candidate edges representing the limits of the vehicles currently circulating on the road. Thus, a symmetry map [3] of the ROI is computed so as to enhance those objects in the scene that present strong colour symmetry characteristics. After that, vertical edges are considered in pairs around those regions of the ROI where a sufficiently high symmetry measure (rejecting uniform areas) has been obtained, in order to account only for couples that represent possible vehicle contours, disregarding those combinations that lead to unrealistic vehicle shapes. Only those regions complying with a realistic edges structure will be validated as candidate vehicles.

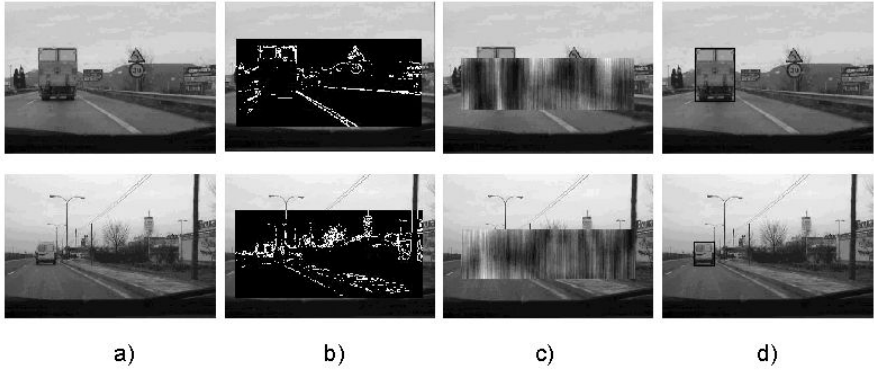


Fig. 2. Vehicle detection examples. a) Original image. b) ROI edge enhancement. c) Symmetry map. d) Position of detected vehicle.

Temporal consistence. Using spatial features as the only criterion for detecting obstacles yields to sporadic incorrect vehicles detection in real situations due to noise. Hence, a temporal validation filter becomes necessary to remove non-consistent objects from the scene [14]. This means that an object validated under the spatial features criterion described in the previous section must be detected several consecutive iterations in the image in order to be considered as a real vehicle. Otherwise it is discarded. A value $t = 0.5s$ has been used in practice to ensure that a vehicle appears in the image in a consistent time sequence. During the time-spatial validation stage a major problem is to identify the appearance of the same vehicle in two consecutive frames. For this purpose, its (x, y) position in correlative frames is used to keep track of temporal consistence of all candidate vehicles. Figure 2 depicts a couple of examples where the original and filtered images are illustrated together with the symmetry map of the ROI and the final position of the detected vehicle.

3.3 Vehicle Tracking

The position of the vehicle detected in the previous stage is tracked in two steps: position measurement and position estimation. The image contained inside the bounding box of the vehicle detected in the previous iteration is used as template in order to detect the updated position of the vehicle in the current image basing on a best fit correlation. After that, data association for position validation is carried out using the (x, y) location of the newly detected vehicle. Basically it must be determined whether some of the objects in the current frame corresponds to the vehicle under tracking. For this purpose, a limited searching area is specified around the vehicle position yielding to efficient and fast detection. Likewise, a minimum correlation value and template size are established so as to determine the end of the tracking process, whenever poor

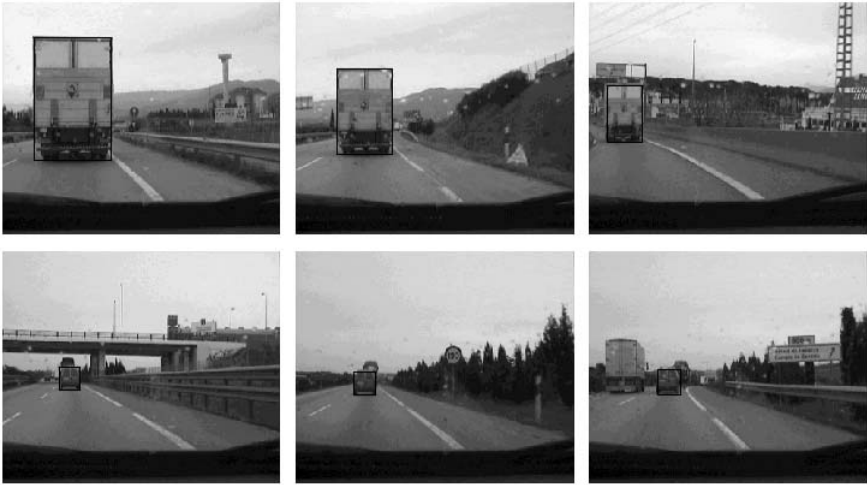


Fig. 3. Examples of vehicle tracking in real traffic situations.

correlations are attained or in case the vehicle gets too far away or out of the scene. The vehicle position measurements are then filtered using a recursive least squares estimator with exponential decay [13]. To avoid the problem of partial occlusions, the previously estimated vehicle position is kept during 5 consecutive iterations without obtaining any validated position, before considering that the vehicle track has been lost. If this happens, vehicle tracking is stopped and the vehicle detection stage is started again. To illustrate the vehicle tracking algorithm, figure 3 shows some real traffic situations in which the preceding vehicle position is tracked in a sequence of images.

3.4 Adaptive Navigation

Upon detecting the position of the preceding vehicle proper actions must be taken in order to ensure safe navigation, in an adaptive cruise control manner. Thus, if the time separation between the preceding vehicle and the ego-vehicle is below some predefined safety interval (2s in this work) the ego-vehicle velocity is accordingly modified. Further navigation actions can also be undertaken in case the preceding vehicle performs a sudden braking manoeuvre in the limits of the safety interval. This could lead to an imminent crash unless the ego-vehicle rapidly detects the braking situation and reacts accordingly producing also a fast braking. In order to achieve this goal, the activation of the preceding vehicle braking lights must be detected as it will clearly indicate a braking down manoeuvre. The position of braking lights changes depending on each vehicle model and company, and thus, a detailed search should be carried out so as to accurately locate them inside the vehicle bounding box. Nonetheless, there is



Fig. 4. Detection of a sudden braking of the preceding vehicle by continuously monitoring the activation of the braking lights.

some a priori information that can obviously be used to enhance and ease the searching process, as long as braking indicators are usually two red lights symmetrically located near the left and right sides of the rear part of the vehicle. This constraint will help locate the braking lights for different types of vehicles. The next step implies to detect a sudden activation of the braking lights, as previously mentioned, by continuously monitoring their luminance component. In case this occurred, an alarm would be raised to the vehicle navigator so as to produce an emergency braking down manoeuvre. Figure 4 depicts an example where a sudden braking of the preceding vehicle is detected following the previously described scheme.

4 Traffic Sign Detection

The vision based traffic sign detection module developed in this work manages 289x352 colour images in RGB (Red, Green, Blue) format. As in the previous sections, a ROI is defined intended to reduce the processing time by focusing on the area of the image where traffic signs are more likely to appear. For this purpose, the 10% upper part and the 30% lower part of the image are disregarded for visual computations. On the other hand, appropriate choice of the colour features to use in the process is of crucial importance in order to attain proper and fast detection. Accordingly, only the Red component is considered as it provides a high capacity for colour discriminating in the visual analysis of traffic signs, and no further preprocessing is needed after digitalisation. In an attempt to carry out a preattentive strategy, a coarse analysis of vertical edges is performed in a first stage basing on differential characteristics computed on the Red component of the image using a gradient filter, as depicted in figure 5, where several images containing traffic signs are illustrated together with their associated gradient images after applying a vertical edge operator.

The accumulated projection of the vertical edges computed upon gradient images provides quite a valuable and useful information for discriminating the positions of the image where traffic signs are located, as can be derived from observation of figure 6. Traffic signs can be clearly characterized by the presence

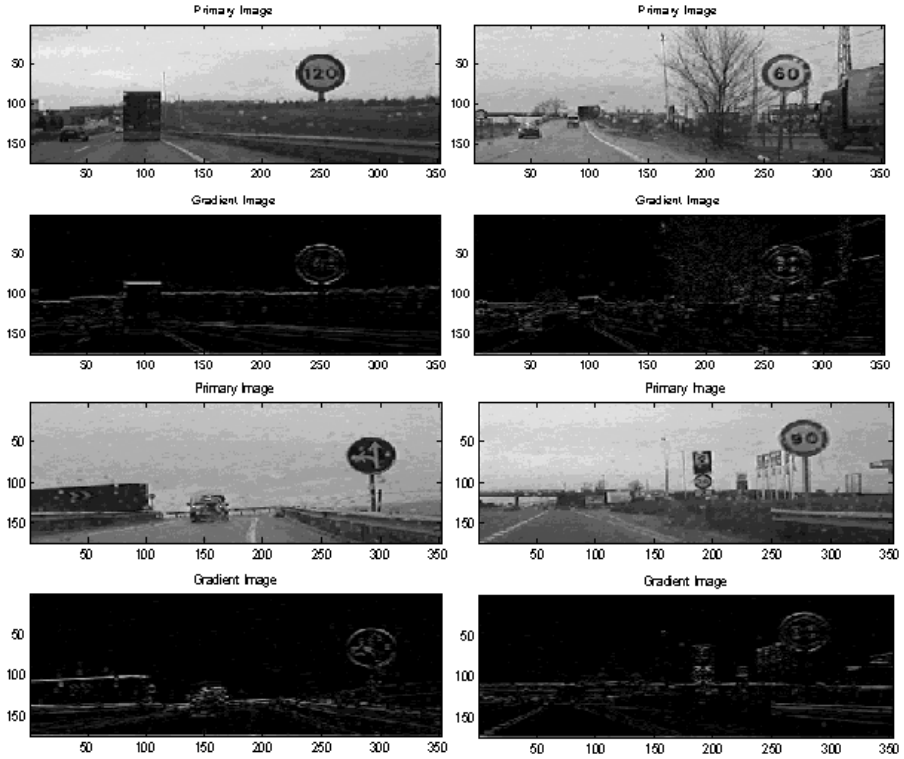


Fig. 5. Original and filtered images using a vertical gradient operator.

of two consecutive peaks of high value very close to each other in the projection profile. This fact constitutes the basis for the traffic sign detection algorithm proposed in this work. As a first step, an adaptive thresholding is performed aiming at removing the common offset component in the projection profile. For this purpose, a threshold u is computed as expressed in equation 4.

$$u = \mu + \mu^+ \quad (4)$$

where μ stands for the average value of the projection profile, while μ^+ represents the average of all points in the projection profile whose value is greater than μ . The resulting threshold is depicted in figure 6 for the four edge images shown in figure 5. As can be appreciated, the peaks corresponding to traffic signs positions remain after thresholding. Finally, the coarse detection phase ends by removing narrow peaks from the projection profile. This yields a set of candidate image regions that highly reduces and constraints the portions of the image where traffic signs are likely to appear, as depicted in figure 7. A further analysis should be accomplished so as to validate the existence of circular or triangular shapes within the candidate regions, prior to the traffic sign recognition stage.

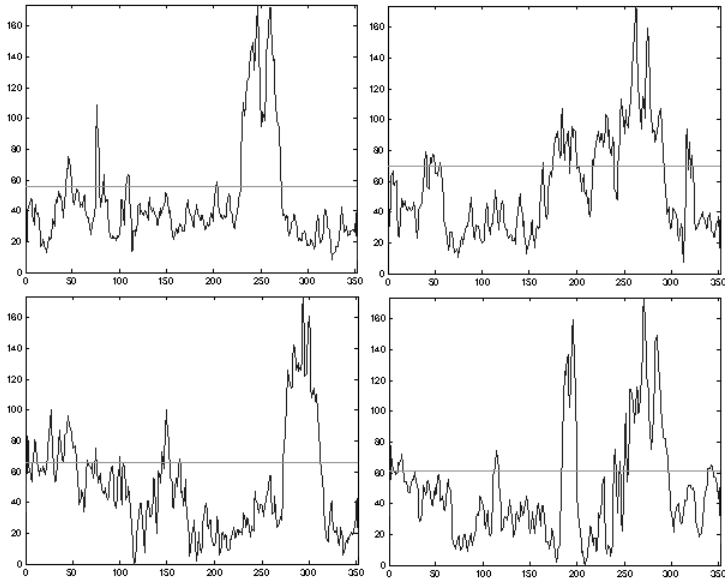


Fig. 6. Adaptive threshold u computed over the projection profile.

The algorithm described in this section has been implemented on a PentiumIV running at 2.5GHz, yielding an average execution time of 106ms.

5 Discussion

The complete vision and DGPS based system for autonomous and/or assisted navigation described in this paper has been implemented on the so-called Babieca prototype vehicle, an electric commercial Citroen Berlingo, as a first step towards its long-term deployment on urban scenarios. Practical experiments were conducted on a private circuit located at the *Industrial Automation Institute* in Arganda del Rey (Madrid) to test the autonomous navigation system. The circuit is composed of several stop stations, streets, intersections, and roundabout points, trying to emulate an urban quarter. Babieca ran over hundreds of kilometres in lots of successful autonomous missions carried out along the test circuit. Besides, a live demonstration exhibiting the system capacities on autonomous navigation was carried out during the IEEE Conference on Intelligent Vehicles 2002, in a private circuit located at Satory (Versailles), France. A complete set of video files demonstrating the operational performance of the system in real tests can be retrieved from <ftp://www.depeca.uah.es/pub/vision>. Practical trials have also been carried out on real urban and highway scenarios so as to test the validity of the vision based driving assistance systems developed in this work. Either the vehicle detection or the traffic sign module have proven to be useful for the

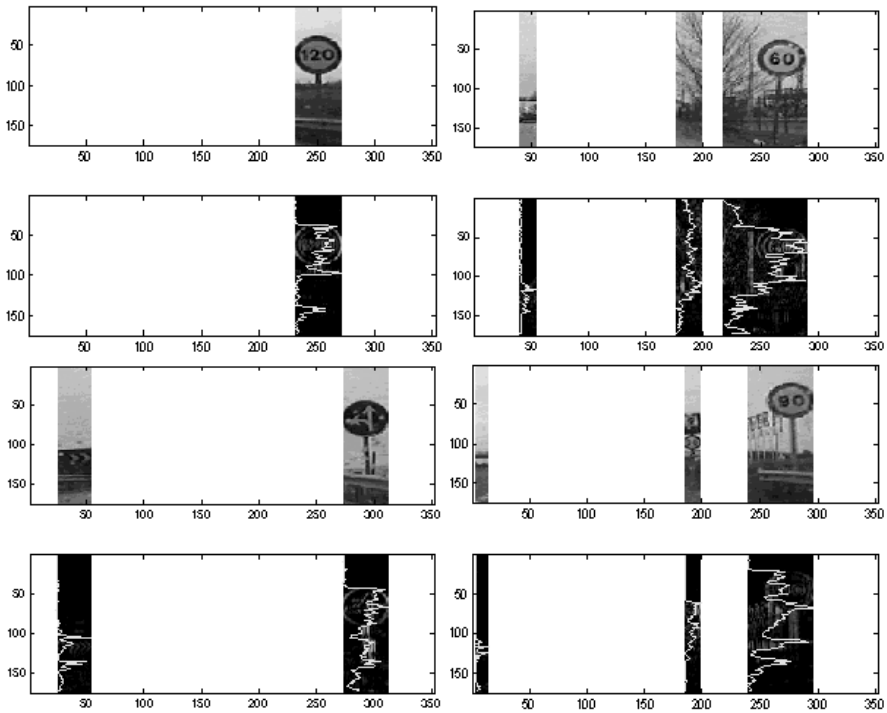


Fig. 7. Candidate image regions where traffic signs are likely to be located.

driver in keeping a safety distance and fulfilling the traffic rules, respectively, in real conditions. On the other hand, the integration of the vehicle detection task into the global navigation system has been successfully achieved, as demonstrated in the previously mentioned video files. The fact that no extremely high precision is needed in the DGPS signal, together with the use of a single colour camera, results in a low cost final system suitable for midterm commercial development. Our current work focuses on the detection of pedestrians using a combination of vision and laser sensors, as it would be a major issue in increasing traffic safety.

Acknowledgements. This work is supported by the University of Alcalá (project UAH2002/031) and the CICYT (project DPI2002-04064-C05-04), as well as by the generous support of the Instituto de Automática Industrial del Consejo Superior de Investigaciones Científicas.

References

1. M. Bertozzi and A. Broggi.: GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*. **7** (1), (1998). 62–81.
2. M. Bertozzi, A. Broggi and A. Fascioli.: Vision-based intelligent vehicles: State of the art and perspectives. *Robotics and Automation Systems* **32** 2000. 1–16.
3. A. Broggi, S. Nichele, M. Bertozzi, and A. Fascioli.: Stereo Vision-based Vehicle Detection. In *Proceedings of the IEEE Intelligent Vehicles Symposium*. Detroit, USA. October (2000)
4. P. Charbonnier, P. Nicolle, Y. Guillard and J. Charrier.: Road boundaries detection using color saturation. *Proceedings of the Ninth European Signal Processing Conference '98*. September (1998)
5. T. De Pedro, R. Garcia, C. Gonzalez, J. E. Naranjo, J. Reviejo and M. A. Sotelo.: Vehicle Automatic Driving System Based on GNSS. *Proceedings of the International Conference on Intelligent Vehicles*. Seville, (2001)
6. E. D. Dickmanns, R. Behringer, D. Dickmanns, T. Hildebrant, M. Mauer, F. Thomanek, and J. Shiellhlen.: The seeing passenger car 'VaMoRs-P'. In *Proc. of Int. Symp. on Intelligent Vehicles*, Paris. (1994)
7. U. Franke, D. Gavril, S. Gorzig, F. Lindner, F. Paitzold and C. Wohler.: Autonomous driving goes downtown. *Proceedings of the IEEE Intelligent Vehicles Symposium*. Stuttgart, Germany. (1998)
8. A. Giachetti, M. Campani and V. Torre.: The use of Optical Flow for Road Navigation. *IEEE Transactions on Robotics and Automation*. **14** No 1, (1998)
9. R. C. Gonzales and R. E. Wood.: *Digital Image Processing*. Addison-Wesley, Reading, MA, (1992)
10. N. Ikonomakis, K. N. Plataniotis and Venetsanopoulos.: Color Image Segmentation for Multimedia Applications. *Journal of Intelligent and Robotic Systems*. (2000)
11. W. Li, X. Jiang and Y. Wang.: Road recognition for vision navigation of an autonomous vehicle by fuzzy reasoning. *Fuzzy Sets and Systems*. **93** Elsevier. (1998)
12. R. Gregor, M. Lutzeler, M. Pellkofer, K. H. Siedersberger, and E. D. Dickmanns.: EMS-Vision: A Perceptual System for Autonomous System. *IEEE Transactions on Intelligent Transportation Sustems*. **3**, No. 1, (2002)
13. H. Schneiderman and M. Nashman.: A Discriminating Feature Tracker for Vision-Based Autonomous Driving. *IEEE Transactions on Robotics and Automation*. **10** NO 6. (1994)
14. M. A. Sotelo.: *Sistema de Navegación Global Aplicado al Guiado de un Vehículo Autónomo Terrestre en Entornos Exteriores Parcialmente Conocidos*. Phd Thesis Dissertation. University of Alcalá. (2001).
15. D. F. Wolfram Burgard and S. Thrun.: *Active Markov Localization for Mobile Robots*. Elsevier Preprint. (1998)

Using Fractional Calculus for Lateral and Longitudinal Control of Autonomous Vehicles^{*}

J.I. Suárez¹, B.M. Vinagre¹, A.J. Calderón¹, C.A. Monje¹, and Y.Q. Chen²

¹ EII, Universidad de Extremadura, Badajoz - Spain
{jisuares,bvinagre,ajcalde,cmonje}@unex.es

² CSOIS, Utah State University, Logan, Utah - USA
yqchen@ece.usu.edu

Abstract. Here it is presented the use of Fractional Order Controllers (FOC) applied to the path-tracking problem in an autonomous electric vehicle. A lateral dynamic model of a industrial vehicle has been taken into account to implement conventional and Fractional Order Controllers. Several control schemes with these controllers have been simulated and compared. First, different controllers with similar parameters have been implemented and then they have been improved by using optimization methods. The preliminary results are presented here.

1 Introduction

Path-tracking problems in autonomous vehicles and mobile robotics have been investigated for the last two decades. Some methods proposed can be basically divided into *temporal* (based on the application of control theory) and *spatial* controllers (based on geometric methods such as *pure-pursuit*) (see [1] and [2] for additional references). In this work, a spatial path tracking method, called the ϵ -controller, is applied which was first proposed in [3]. This path tracking method computes the normal distance from the vehicle to the desired path, ϵ , and generates a desired velocity vector for the vehicle to follow the path.

To improve its performance, several regulation schemes using the fractional-order control (FOC) idea [4] have been investigated here. FOC is based on “fractional-order calculus”. Recent books [5,6,7,8] provide a good source of references on fractional-order calculus. However, applying fractional order calculus to dynamic systems control is just a recent focus of interest [9,10,11,12,13,14]. For pioneering works, we cite [4,15,16,17]. For the latest development of fractional calculus in automatic control and robotics, we cite [18]. As for path tracking problems, the first experiences in path-tracking applied to XY cutting tables and mobile robotics can be found in [19] and [20].

In this paper we present some preliminary results of the use of FOC in path-tracking problems applied to an autonomous electric vehicle by using an ϵ -controller on path-tracking basic algorithm.

^{*} This work has been partially supported by Research Grant DPI 2002-04064C05-03 (MCYT).

The rest of the paper is organized as follows: in Sec. 2 an introduction to fractional calculus is made. In Sec. 3 the lateral dynamic model of the vehicle is shortly described. In Sec. 4 the ϵ -controller is briefly introduced. Section 5 presents several control schemes (P, PI, PI^α , ...) with a special attention to FOC. Section 6 shows some of our simulated results. Finally, in Sec. 7, some conclusions are outlined.

2 Introduction to Fractional Calculus

Even though the idea of fractional order operators is as old as the idea of the integer order ones is, it has been in the last decades when the use of fractional order operators and operations has become more and more popular among many research areas. The theoretical and practical interest of these operators is nowadays well established, and its applicability to science and engineering can be considered as emerging new topics. Even if they can be thought of as somehow ideal, they are, in fact, useful tools for both the description of a more complex reality, and the enlargement of the practical applicability of the common integer order operators. Among these fractional order operators and operations, the fractional integro-differential operators (fractional calculus) are specially interesting in automatic control and robotics.

2.1 Fractional Order Operators

Fractional calculus is a generalization of integration and differentiation to non-integer (fractional) order fundamental operator ${}_aD_t^\alpha$, where a and t are the limits and α , ($\alpha \in \mathbb{R}$) the order of the operation. The two definitions used for the general fractional integro-differential are the Grünwald-Letnikov (GL) definition and the Riemann-Liouville (RL) definition. The GL definition is that

$${}_aD_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t-a}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t-jh) \quad (1)$$

where $\lfloor \cdot \rfloor$ means the integer part, while the RL definition is

$${}_aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \quad (2)$$

for $(n-1 < \alpha < n)$ and where $\Gamma(\cdot)$ is the Euler's *gamma* function.

For convenience, Laplace domain notion is usually used to describe the fractional integro-differential operation. The Laplace transform of the RL fractional derivative/integral (2) under zero initial conditions for order α , ($0 < \alpha < 1$) is given by

$$\mathcal{L}\{{}_aD_t^{\pm\alpha} f(t); s\} = s^{\pm\alpha} F(s). \quad (3)$$

2.2 Fractional Order Control Systems

In theory, the control systems can include both the fractional order dynamic system to be controlled and the fractional-order controller. A fractional order plant to be controlled can be described by a typical n -term linear FODE in time domain

$$a_n D_t^{\beta_n} y(t) + \cdots + a_1 D_t^{\beta_1} y(t) + a_0 D_t^{\beta_0} y(t) = 0 \quad (4)$$

where $a_k (k = 0, 1, \dots, n)$ are constant coefficients of the FODE; $\beta_k, (k=0, 1, 2, \dots, n)$ are real numbers. Without loss of generality, assume that $\beta_n > \beta_{n-1} > \cdots > \beta_1 > \beta_0 \geq 0$. Consider a control function which acts on the FODE system (4) as follows:

$$a_n D_t^{\beta_n} y(t) + \cdots + a_1 D_t^{\beta_1} y(t) + a_0 D_t^{\beta_0} y(t) = u(t). \quad (5)$$

By Laplace transform, we can get a fractional transfer function :

$$G_p(s) = \frac{Y(s)}{U(s)} = \frac{1}{a_n s^{\beta_n} + \cdots + a_1 s^{\beta_1} + a_0 s^{\beta_0}}. \quad (6)$$

In general, a fractional-order dynamic system can be represented by a transfer function of the form:

$$G_p(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\alpha_m} + \cdots + b_1 s^{\alpha_1} + b_0 s^{\alpha_0}}{a_n s^{\beta_n} + \cdots + a_1 s^{\beta_1} + a_0 s^{\beta_0}} \quad (7)$$

However, in control practice, more common is to consider the fractional order controller. This is due to the fact that the plant model may have already been obtained as an integer order model in classical sense. In most cases, our objective is to apply the fractional order control (FOC) to enhance the system control performance. Taking conventional PID-controller as an example, its fractional order version, $PI^\lambda D^\mu$ controller, was studied in time domain and in frequency domain. The time domain formula is that

$$u(t) = K_p e(t) + T_i D_t^{-\lambda} e(t) + T_d D_t^\mu e(t). \quad (D_t^{(*)} \equiv 0 \ D_t^{(*)}). \quad (8)$$

It can be expected that $PI^\lambda D^\mu$ controller (8) may enhance the systems control performance due to more tuning knobs introduced.

3 Vehicle Dynamic Model

The vehicle used in our work is a Citroën Berlingo with an Ackerman steering system [21]. For modeling the lateral dynamics of the vehicle a body-fixed coordinate systems (BFCS) is fixed to its center of gravity (CG) and the roll, pitch, bounce and deceleration dynamics are neglected. A linear model can be obtained by solving the dynamic equations and further simplifications are done [22]. We assume for simplicity that the both front wheel turn the same amount of angle and hence each wheel produces the same steering forces. The resulting model is known as the bicycle dynamics model. Figure 1 shows a diagram where the main parameters and variables are depicted. They are the following ones:

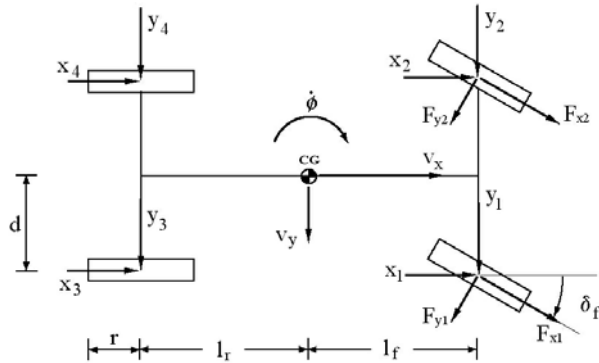


Fig. 1. Ackerman steered vehicle and its related forces for the lateral dynamic model

- v_x, v_y : longitudinal and lateral velocity, respectively.
- δ_f : front wheel steering angle
- $\dot{\phi}$: yaw rate
- m : vehicle mass
- I_z : moment of inertia about the z-axis
- c_f, c_r : front and rear wheel cornering stiffness
- l_f, l_r : distances of the front and rear axles from the CG
- d : distance from car centerline to each wheel (half-track)
- r : wheel radius

The coordinate system uses the convention of the Society of Automotive Engineers with the z-axis pointing into the road and the positive yaw direction as depicted in Fig. 1. The longitudinal velocity v_x is supposed to be approximately constant. Choosing $\dot{\phi}$ and v_y as state variables the vehicle model can be expressed by the following state equations:

$$\begin{bmatrix} \dot{v}_y \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} -2\frac{a_1}{mv_x} - v_x - 2\frac{a_2}{mv_x} \\ -2\frac{a_3}{I_z v_x} - 2\frac{a_4}{I_z v_x} \end{bmatrix} \cdot \begin{bmatrix} v_y \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \delta_f \quad (9)$$

where $a_1 = c_f + c_r$; $a_2 = c_f l_f - c_r l_r = a_3$; $a_4 = c_f l_f^2 + c_r l_r^2$; $b_1 = 2\frac{c_f}{m}$; $b_2 = 2\frac{c_f l_f}{I_z}$.

Parameters for the Citroën Berlingo used in our simulations are: $m = 1466 \text{ Kg}$; $I_z = 28000 \text{ Nm}^2$; $c_f = c_r = 60000 \text{ N/rad}$; $l_f = 1.12 \text{ m}$; $l_r = 1.57 \text{ m}$

4 Path-Tracking Algorithm

The path-tracking problem was accomplished by using the scalar ϵ -controller [3]. Basically, it is a regulator, with the cascade control architecture shown in Fig. 1, that operates on the vehicle normal deviation ϵ from the desired path. The ϵ -controller (C_ϵ) generates a desired velocity vector $\mathbf{V}_I^*$ which depends on the lateral deviation from the path. If the vehicle is near the path the ϵ -controller generates a velocity vector tangent to the path and permits the vehicle to travel

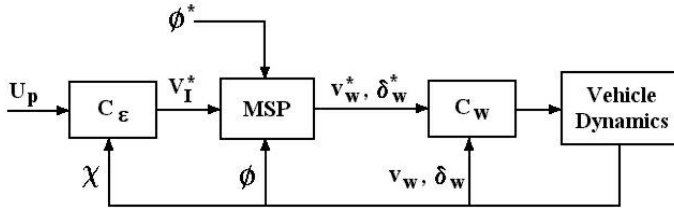


Fig. 2. ϵ -Controller cascade control achitecture

at its maximum speed. However, when the vehicle is far from the desired path, the direction of $\mathbf{V}_I^*$ points to the closest point on the path in the radial direction and then the velocity of the vehicle is reduced.

The *MakeSetPoint* (MSP) algorithm converts the desired velocity vector $\mathbf{V}_I^*$ into body-fixed longitudinal velocity (v_x^*) and steering angle (δ_w^*) setpoints. This is accomplished by rotating the velocity vector, given in an Inertial Cartesian coordinate System (ICS), into the vehicle-fixed coordinate system. Then the longitudinal velocity and the steering angle setpoints are obtained from the vehicle geometry. The low-level controllers (C_w) track the actuator-level setpoints. A detailed description of the algorithm can be found in [3] and [21].

5 Regulation Schemes

The control law of the ϵ -controller turns the two-dimensional path tracking problem into a scalar regulation. It is a nonlinear controller operating on the scalar ϵ , then, several control schemes (P, PI, ...) can be implemented. In the following subsections three different strategies of regulation are presented and compared each other: P, PI and fractional PI^α controllers.

5.1 P Controller

In this regulation method the control signal is proportional to the lateral distance from the path, then

$$u(t) = K_P \epsilon(t) \quad (10)$$

This is a simple regulator, but presents a stationary error. Hence, by increasing the proportional gain (K_P) the vehicle can be stabilized, but the error can not be driven to zero. Furthermore, the bigger the proportional gain, the faster the vehicle turns to path. For these reasons, it is necessary to use other regulators.

This controller has been implemented as a digital P regulator, with a sample period $T = 0.1$ seconds. The transfer function is:

$$G_P(z) = K_P \quad (11)$$

5.2 PI and PI^α Controller

In these kind of controllers the control law can be obtained as follows

$$u(t) = K_P \epsilon(t) + K_I I^\alpha \epsilon(t) \quad (12)$$

being

$$I^\alpha f(t) \equiv \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau) d\tau, \quad t > 0, \quad \alpha \in \mathbb{R}^+ \quad (13)$$

With $1 < \alpha < 2$ we have a fractional controller. However, when $\alpha = 1$, (12) becomes

$$u(t) = K_P \epsilon(t) + K_I \int_0^t \epsilon(\tau) d\tau \quad (14)$$

which is the control law corresponding to the classical PI controller. In this case, the proportional gain must be lowered to reduced the oscillation and the integral term can be used to drive the error to zero.

The PI controller has been implemented with a digital integrator using the trapezoidal rule. The transfer function is:

$$G_{PI}(z) = K_P + K_I \frac{T(z+1)}{2(z-1)} \quad (15)$$

In order to preserve the zero stationary error the PI^α controllers have been implemented with two cascade integrators, the first one corresponding $\frac{1}{s}$ and the second one corresponding $\frac{1}{s^{1-\alpha}}$, $\alpha > 1$. The first integrator has been discretized using the Tustin transformation $s = \frac{2(z-1)}{T(z+1)}$. The fractional order integrator $\frac{1}{s^{1-\alpha}}$ has an infinite terms representation and must be discretized using a finite dimensional IIR filter. For our case (see [23]) the method of the Continued Fraction Expansion (CFE) has been used for obtaining a finite dimensional approximation of the fractional power of the Tustin discrete equivalent.

6 Simulations

6.1 Simulink Model

The high-level Simulink model shown in Fig. 3 simulates the control architecture depicted in Fig. 2. The *Behavior Generator* subsystem outputs the parameters of the path geometry which are used for the *Sensor Motion Scheduler (SMS)* to provide all the quantities required by the ϵ -controller, in order to keep the vehicle on the desired path. The SMS is implemented together with the ϵ -controller and the MSP algorithm in the second subsystem, whose outputs are the desired velocity and steering angle. The next subsystem includes the steering system, that is modeled as a first order system with a steering angle limitation of 30° , and a low-level PI controller. The two next blocks are the dynamic lateral vehicle model and a block that computes the vehicle state. Finally, in the feedback path a sensor suite provides only the necessary state variables to the path-tracking controllers.

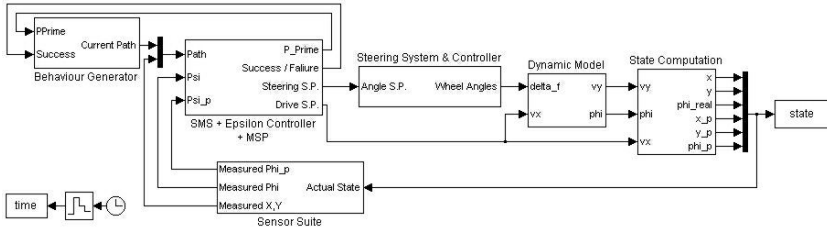


Fig. 3. Simulink model of the vehicle with the ϵ -controller

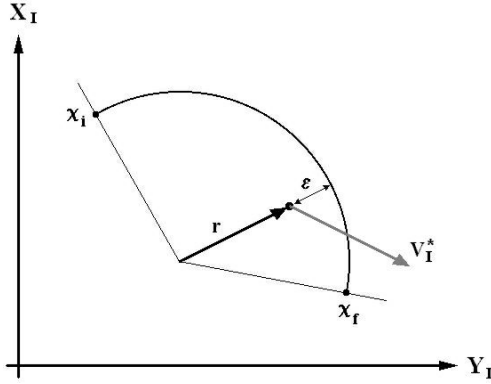


Fig. 4. ϵ -Controller parameters for an arc segment

6.2 First Results

The firsts simulations have been done to compare the performance of several kind of controllers. In these simulations the vehicle has been supposed to describe a semicircular path. First, the input vector to C_ϵ block $\mathbf{U}_P = [\chi_i \ \chi_f \ r \ V_d]^T$ shown in Fig. 2 must be established, where χ_i and χ_f are the initial and final points, respectively, r is the radius of the path and V_d is desired vehicle velocity (Fig. 4). The parameters chosen in our simulations are: $\chi_i = (0, 0)$, $\chi_f = (0, 50)$, $r = 25m$ and $V_d = 10m/s$. The starting point of the vehicle is $(0, 0)$, then the vehicle describes, at the maximum speed of $10m/s$, a semicircular path with radius of 25 metres. In the present case, velocity is not important, for this reason when the vehicle is far from the path the ϵ -controller will decrease the velocity and when it is on the path the vehicle will travel at its maximum speed ($10m/s$).

Five control schemes have been simulated: P, PI, and three FOC, one $PI^{1.5}$ and two $PI^{1.25}$ controllers with different values of K_P . Table 1 shows the parameters used in our simulations for the different controllers. Note that K_P is the same for all the controllers, except for the last one, in which the proportional gain has been increased to show the effects of varying K_P in the fractional controllers. Moreover, all the PI controllers have the same integral gain K_I . The aim of these simulations is to demonstrate that with a controller with an additional tuning parameter a better result can be obtained. It can be thought that a

Table 1. Controllers Parameters

<i>Controllers</i>	K_P	K_I	α
P	20	0	0
PI	20	5	1
$PI^{1.5}$	20	5	1.5
$PI^{1.25} (1)$	20	5	1.25
$PI^{1.25} (2)$	22	5	1.25

Table 2. Computed ISE

P	PI	$PI^{1.5}$	$PI^{1.25} (1)$	$PI^{1.25} (2)$
0.0752	0.0478	0.0558	0.0541	0.0436

better performance can be reached with a PI controller, with two tuning parameters, than with a P controller with only one tuning parameter. The same consideration can be done with a PI^α controller an PI controller.

Figure 5 shows the responses of three tested controllers: P, PI and $PI^{1.25}$ with $K_P = 22$. A positive value of ϵ means the vehicle lies inside the desired path, whereas a negative one means the vehicle is outside the arc. Note the steady state error of the P controller. As explained before, the integral part of the PI controllers tends to eliminate this error.

Figure 6 shows the behavior for the three fractional controllers. For classical PI controller we only can vary two parameters (K_P and K_I). Note that in this case, for FOC an additional parameter (α) can be varied to obtain different responses.

With the aim of comparing the regulation schemes we have computed the integral squared error (ISE) given by:

$$e = \sum_{i=0}^N \epsilon_i^2 \tag{16}$$

being this error the distance between the desired path and the actual path that the vehicle travels. Table 2 shows the computed ISE for the different controllers. Note the smaller area for the PI controllers; as it was obvious, by reducing the steady state error, the ISE will be reduced too. PI and PI^α controllers present a better performance than P controller and the best one is the $PI^{1.25}(2)$.

6.3 Optimized Results

In the next simulations the previous controllers (P, PI and PI^α) have been optimized and four comparison parameters have been established to test the behavior of the controllers. They are:

- the *integral squared error* (ISE), named before,
- the *overshoot*, that it is the maximun deviation from the desired path,

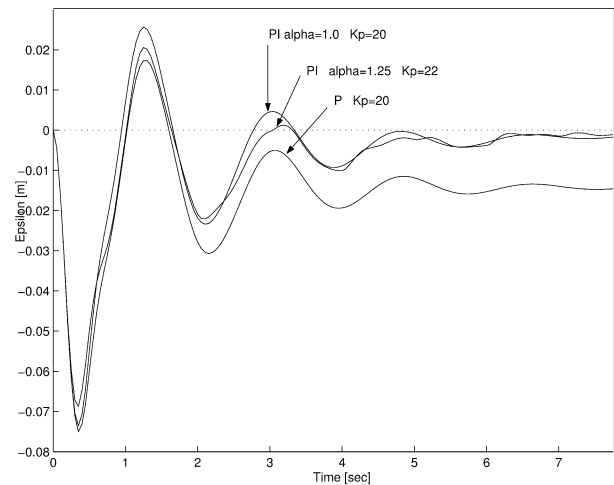


Fig. 5. Deviation ϵ along the path for P, PI and PI^{1.25} controllers

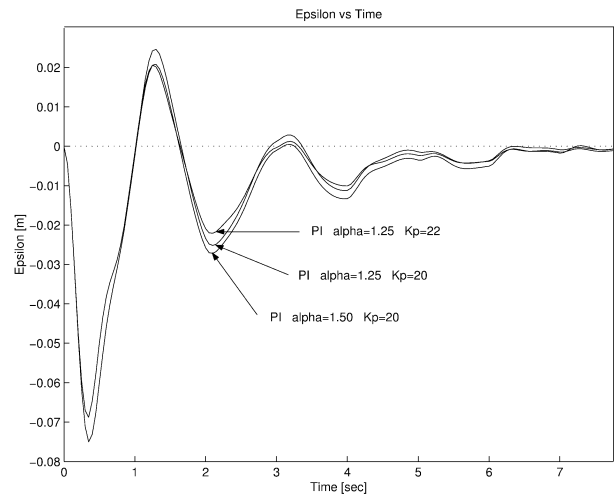


Fig. 6. Deviation ϵ along the path for different FOC

- the *settling time*, defined as the time spent to reach and keep inside 0.01 meters from the path,
- and the *first zero-crossing time*, the time to reach to the first zero-crossing or also the first time to reach $\epsilon = 0$.

The optimization of the parameters of the controllers has been done by minimizing the ISE. The optimized parameters are shown in Table 3.

In Table 4 it can be observed that the best performance corresponds to the fractional order controller. It has the least ISE, settling time and first zero-crossing time. As it was expected, with an additional parameter the best results

Table 3. Controllers Optimized Parameters

<i>Controllers</i>	K_P	K_I	α
P	42.6934	0	0
PI	38.9035	21.6490	1
PI^α	40.0267	20.1579	1.1217

Table 4. Comparison Parameters

<i>Controllers</i>	ISE	Ov	t_{set}	t_{zc}
P	0.017573	-0.0387	4.0535	0.7822
PI	0.011303	-0.0409	2.0535	0.7767
PI^α	0.011258	-0.0403	2.0101	0.7730

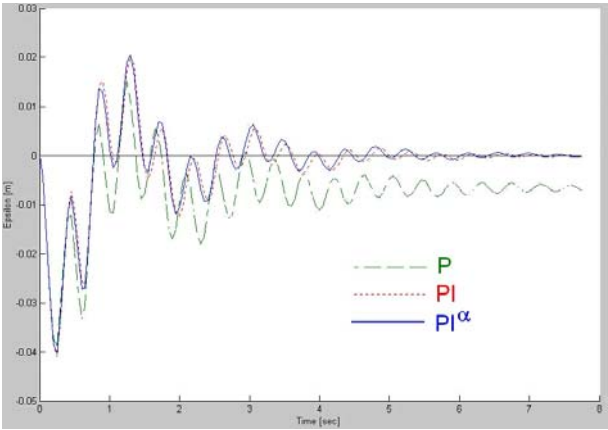


Fig. 7. Controllers responses with their optimal values

corresponds to the controller with more number of parameters. It can be observed in the fractional controller that with a greater α , the proportional gain increases and the integral gain decreases. These results are depicted in Fig. 7. The PI and PI^α controllers have a similar behavior, but the fractional controller is better.

7 Conclusions

Firstly, several controllers have been compared to show their performance in path-tracking problems using a dynamic model of a Citro  n Berlingo vehicle. A novel and simple regulation scheme, fractional PI^α controller, was simulated with better results than the other regulations schemes (P and PI controllers). A sintonization method for the controllers has been presented with the aim of improving their performances and obtaining the optimal values for the parameters K_P , K_I and α . The fractional calculus has been demonstrated to be another useful tool for controllers design. New sintonization methods will be looked for in

future works, and also low level fractional order controllers will be implemented to improve the vehicle performance.

References

1. Ollero, A.: *Robotica. Manipuladores y Robots Moviles*. Marcombo (2001)
2. Wit, J.S.: *Vector Pursuit Path Tracking for Autonomous Ground Vehicles*. PhD thesis, University of Florida (2000)
3. Davidson, M., Bahl, V.: The Scalar ϵ -Controller: A Spatial Path Tracking Approach for ODV, Ackerman, and Differentially-Steered Autonomous Wheeled Mobile Robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation*. (2001) 175–180
4. Manabe, S.: The Non-Integer Integral and its Application to Control Systems. *JIEE (Japanese Institute of Electrical Engineers) Journal* **80** (1960) 589–597
5. Oldham, K.B., Spanier, J.: *The Fractional Calculus*. Academic Press, New York (1974)
6. Samko, S.G., Kilbas, A.A., Marichev, O.I.: *Fractional Integrals and Derivatives and Some of Their Applications*. Nauka i tehnika, Minsk (1987)
7. Miller, K.S., Ross, B.: *An Introduction to the Fractional Calculus and Fractional Differential Equations*. Wiley, New York (1993)
8. Podlubny, I.: *Fractional Differential Equations*. Volume 198. Academic Press (1999)
9. Lurie, B.J.: Three-Parameter Tunable Tilt-Integral-Derivative (TID) Controller. US Patent US5371670 (1994)
10. Podlubny, I.: Fractional-Order Systems and $PI^\lambda D^\mu$ -Controllers. *IEEE Trans. Automatic Control* **44** (1999) 208–214
11. Oustaloup, A., Mathieu, B., Lanusse, P.: The CRONE Control of Resonant Plants: Application to a Flexible Transmission. *European Journal of Control* **1** (1995)
12. Oustaloup, A., Moreau, X., Nouillant, M.: The CRONE Suspension. *Control Engineering Practice* **4** (1996) 1101–1108
13. Raynaud, H., Zergalnoh, A.: State-Space Representation for Fractional Order Controllers. *Automatica* **36** (2000) 1017–1021
14. Vinagre, B.M., Petráš, I., Podlubny, I., Chen, Y.Q.: Using Fractional Order Adjustment Rules and Fractional Order Reference Model in Model-Reference Adaptive Control. *Nonlinear Dynamics* **29** (2002) 269–279
15. Manabe, S.: The non-Integer Integral and its Application to Control Systems. *ETJ of Japan* **6** (1961) 83–87
16. Oustaloup, A.: Fractional Order Sinusoidal Oscillators: Optimization and Their Use in Highly Linear FM Modulators. *IEEE Transactions on Circuits and Systems* **28** (1981) 1007–1009
17. Axtell, M., Bise, E.M.: Fractional calculus applications in control systems. In: *Proceedings of the IEEE 1990 Nat. Aerospace and Electronics Conf.*, New York, USA (1990) 563–566
18. Vinagre, B.M., Chen, Y.: *Lecture Notes on Fractional Calculus Applications in Automatic Control and Robotics*. The 41st IEEE CDC2002 Tutorial Workshop #2, <http://mechatronics.ece.usu.edu/foc/cc02tw/cdrom/lectures/book.pdf>, or, <http://eii.unex.es/isa/LasVegas/> (2002)

19. Orsoni, B., Melchior, P., Oustaloup, A.: Davidson-Cole Transfer Function in Path Tracking Design. In: Proceedings of the 6th European Control Conference, Porto, Portugal (2001) 1174–1179
20. Orsoni, B., Melchior, P., Oustaloup, A., Badie, T., Robin, G.: Fractional motion control: Application to an xy cutting table. *Nonlinear Dynamics* **29** (2002) 297–314
21. Bahl, V.: Modeling and Control of a Class of Autonomous Wheeled Mobile Robots. Master's thesis, Department of Electrical and Computer Engineering, Utah State University, Logan, Utah (2002)
22. Brennan, S.: Modeling and control issues associated with scaled vehicles. Master's thesis, University of Illinois (1999)
23. Vinagre, B.M., Podlubny, I., Hernandez, A., Feliu, V.: Some Approximations of Fractional Order Operators Used in Control Theory and Applications. *Fractional Calculus and Applied Analysis* **3** (2000) 231–248

Recent Advances in the Walking Tree Method for Biological Sequence Alignment

Paul Cull and Tai Hsu

Computer Science Department,
Oregon State University,
Corvallis, OR 97331
`{pc,hsuta}@cs.orst.edu`

Abstract. The meaning of biological sequences is a central problem of modern biology. Although string matching is well-understood in the edit-distance model, biological strings with transpositions and inversions violate this model's assumptions. To align biologically reasonable strings, we proposed the Walking Tree Method [4,5,6,7,8]; an approximate string alignment method that can handle insertion, deletions, substitutions, translocations, and more than one level of inversions. Our earlier versions were able to align whole bacterial genomes (1 Mbps) and discover and verify genes. As extremely long sequences can now be deciphered rapidly and accurately without amplification [2,3,15], speeding up the method becomes necessary. Via a technique that we call recurrence reduction in which some computations can be looked up rather than recomputed, we are able to significantly improve the performance, e.g. 400% for a 1-million base pair alignment. In theory, our method can align a length $|P|$ string with a length $|T|$ string in time $|P||T|/(n \log |P|)$ using n processors in parallel. In practice, we can align 10 Mbps strings within a week using 30 processors.

1 Introduction

Most biological string matching methods are based on the edit-distance model [14] which assumes that changes between strings occur locally. But, evidence shows that large scale changes are possible [9]. In particular, large pieces of DNA can be moved from one location to another (translocations), or replaced by their reversed complements (inversions). Schöniger and Waterman [12] extended the edit-distance model to handle inversions, but their method handles only one level of inversion. Hannenhalli's algorithm [10] for the "translocation" problem runs in polynomial time, but it requires gene locations to be known. Furthermore, it seems unlikely that any simple model will be able to capture the minimum biologically correct distance between two strings. In all likelihood finding the fewest operations that have to be applied to one string to obtain another string will probably require trying all possible sequences of operations. Trying all possible sequences is computationally intractable. This intractability has been confirmed by Caprara [1] who showed that determining the minimum

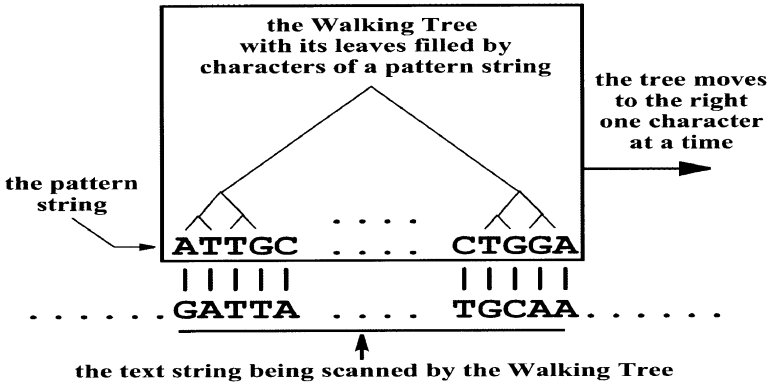


Fig. 1. The picture shows the walking trees structure, a binary tree. Leaves of the tree contain the characters of the pattern string P . After comparing each leaf with a corresponding character of the text string, the walking tree updates its nodes with new scores, then moves to the next position by moving each of its leaves one character to the right. Then it repeats the leaf comparison, and updates its node scores until it reaches the end of the text string.

number of flips needed to sort a sequence is an NP-complete problem, although signed flips can be sorted in polynomial time [11]. We would like a method that runs quickly and can handle insertions, deletions, substitutions, translocations, and inversions. Our Walking Tree heuristic will quickly produce a reasonable alignment of sequences with any combination of these changes [5]. Further, our method can be used to discover genes and to create phylogenies [6]. Here, we describe some recent improvements to the parallel and sequential versions of our method. These improvements allow practical alignments of strings with tens of millions of base pairs.

2 Walking Tree Method

2.1 The Method

The problem is to find an approximate biologically reasonable alignment between two strings, one called pattern P , and the other called text T . Our metaphor is to consider the data structure as a walking tree [19] with $|P|$ leaves, one for each character in the pattern. When the walking tree is considering position $l + 1$, the internal nodes remember some of the information for the best alignment within the first l characters of the text (Figure 1). On the basis of this remembered information and the comparisons of the leaves with the text characters under them, the leaves update their information and pass this information to their parents. The data will percolate up to the root where a new best score is calculated. The tree can then walk to the next position by moving each of its leaves one character to the right. The whole text has been processed when the leftmost leaf of the walking tree has processed the rightmost character of the text.

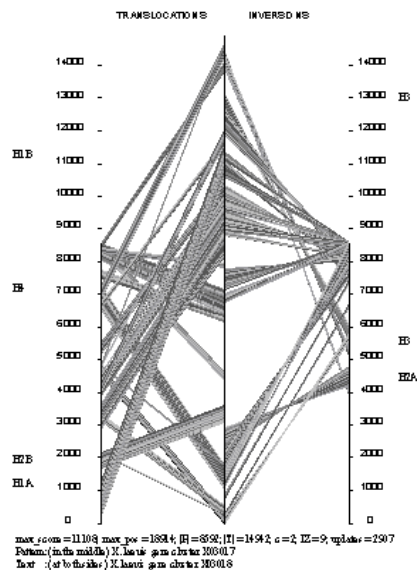


Fig. 2. An alignment of two histone gene cluster from *Xenopus laevis*, Genbank accession number: X03017 (in the middle) and X03018 (at both sides). Note that genes H2A, H2B, H3, and H4 are marked on both sequences. The alignment shows that the orientation of H2A and H3 are reversed in the two sequences. This picture shows the Walking Tree Method is capable of finding inversions and translocations of genes.

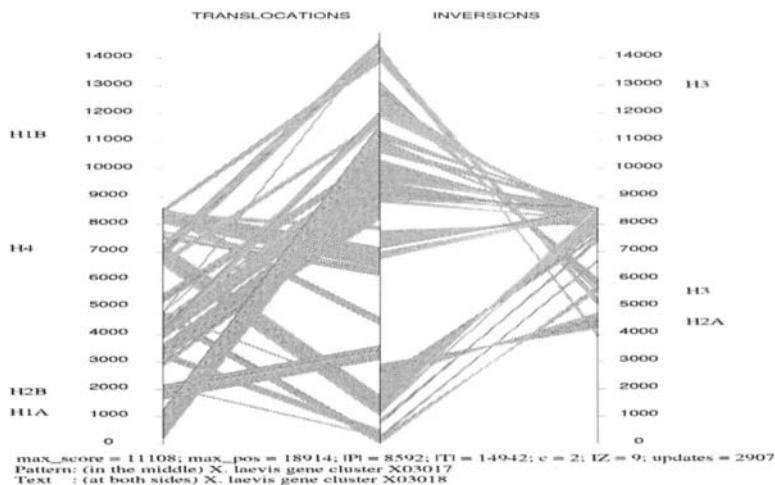


Fig. 3. An alignment of the mitochondrial genomes of *Anopheles quadrimaculatus*, GenBank locus MSQNCATR (in the middle), and *Schizosaccharomyces pombe*, GenBank locus MISPCG (at both sides). The previously unrecognized Cytochrome c oxidase 3 (COX-3) region in this map was identified by the Walking Tree Method.

The alignments in the papers [4,5,6,7] show that the method does handle insertion, deletions, substitutions, translocations, and more than one level of inversions (Figure 2, 3).

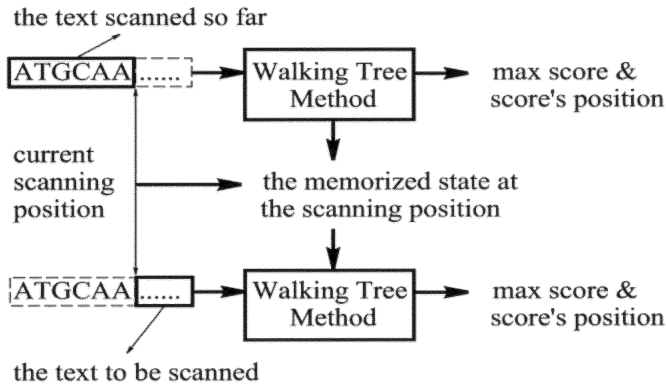


Fig. 4. We use a technique similar to recovering a crashed program by saving its state before crashes. The memorized states record the states of the walking tree and the corresponding scanning positions of the text string. Once we have the recorded information, we can scan the text from the position we have memorized to avoid scanning from the first position of the text.

2.2 Previous Improvement

We recognized that the alignment copying in the original design [4,5,6] was passively activated whenever a better score occurred. It is better to postpone the copying to allow faster scoring at the tree nodes. Based on this idea, we discovered a scalable improvement for both the sequential and the parallel versions of the Walking Tree Method by using a state-caching technique similar to that used in recovering from program crashes (Figure 4). This method recursively calculates and remembers the state of the walking tree at k regularly spaced scanning positions. This memorization results in a sequential algorithm which has $\Theta(k|P||T|)$ runtime using $\Theta((|P| \log |P|)^{1/k})$ space. By making the CPUs spend more time working rather than talking to each other, the parallel version with $\Theta(|P|)$ processors has $\Theta(|T|)$ runtime using $\Theta(|P| \log |P|)$ space.

This improvement allowed us to complete the alignment of two whole genomes of about one million base pairs, *Borrelia burgdorferi* [16] (910724 bps of its single chromosome) and *Chlamydia trachomatis* [13] (1042519 bps). The alignment [8] showed that the Walking Tree Method also works well on long sequences, and the 103 matches found by the Walking Tree Method are consistent with annotations of GenBank (Figure 5, 6).

3 Fast Walking Tree Method

3.1 Why is the New Improvement Needed?

Alignments for sequences of millions of base pairs will become routine, since extremely long sequences can now be deciphered rapidly and accurately by direct, linear analysis of DNA without amplification [2,3,15]. Although our previous improvement is scalable and inexpensive, a million base pair alignment still takes

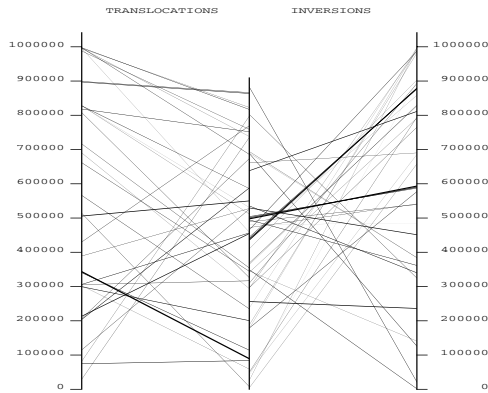


Fig. 5. This picture shows that the Walking Tree Method reveals the matched genes that are labeled (annotated) on both DNAs (the total DNA sequence of *Borrelia burgdorferi* (in the middle) aligned with the total DNA sequence of *Chlamydia trachomatis* (at both sides)). There are 40 translocations and 63 inversions in this picture. Again, this picture shows the Walking Tree Method is capable of finding inversions and translocations of genes that were also annotated in GenBank.

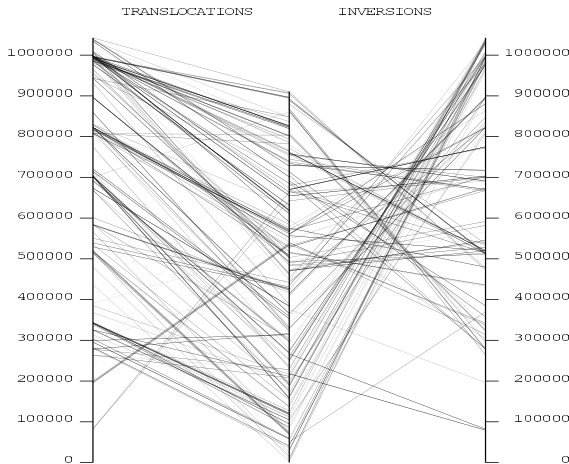


Fig. 6. This picture shows that Walking Tree Method reveals potential genes that are unlabeled on both DNAs. What interests us is the big match (*Chlamydia*: 352764 to 357294, and *Borrelia*: 399872 to 403967) which only covers 50 locus BORRPOB annotated in GenBank database, but is found on both DNAs.

significant time. With this new improvement, such alignments can be done in a couple of hours by an inexpensive network cluster, and alignments of sequences of 32 million base pairs can be done in less than 50 days.

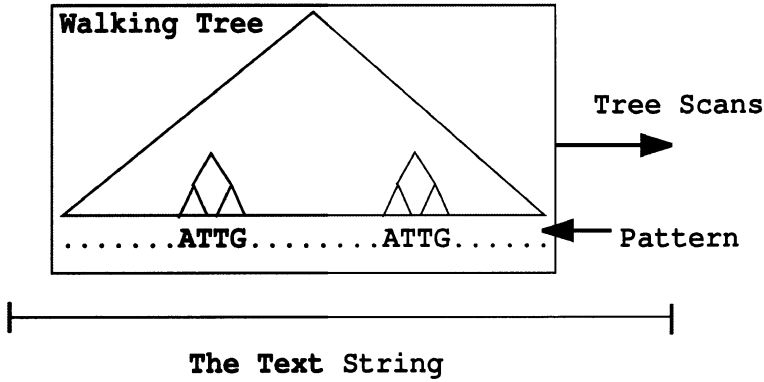


Fig. 7. Both subtrees of the same leaves (labeled with ATTG) have nearly the same results when they scan the same text position.

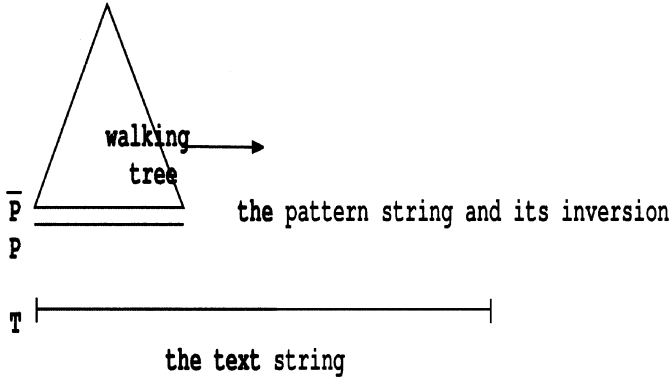


Fig. 8. The original method stores both P and P 's inverse in the leaves of the tree. This design needs $\Theta(|L||T||\Sigma|^{2|L|})$ space and time to compute recurrences for L -leaf subtrees, where Σ is P 's alphabet.

3.2 The Recurrence Idea

The idea behind the fast walking tree method is simply to identify repeated computations, and then eliminate them. This technique which we call Recurrence Reduction is similar to the “Four Russians” technique [17]. To use the technique, we need to identify the recurrent part of the Walking Tree Method.

In the original Walking Tree Method, subtrees with the same leaves produce almost the same results when scanning the same position of the text (Figure 7). So, a subtree can re-use a previously computed result. However, because an original leaf node actually stores information for both the pattern string P and P 's inverse, $\Theta(|L||T||\Sigma|^{2|L|})$ space and runtime are needed to compute all combinations for L -leaf subtrees, where Σ is P 's alphabet (see Figure 8). For **DNA** or **RNA** data, $|\Sigma| = 4$. A further modification is needed to reduce this $\Theta(|L||T||\Sigma|^{2|L|})$ cost.

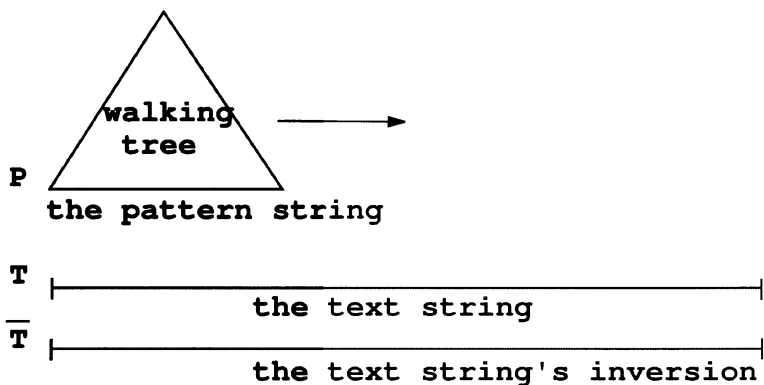


Fig. 9. By reversing the original design of the text and pattern strings, significant space and runtime for recurrences are saved.

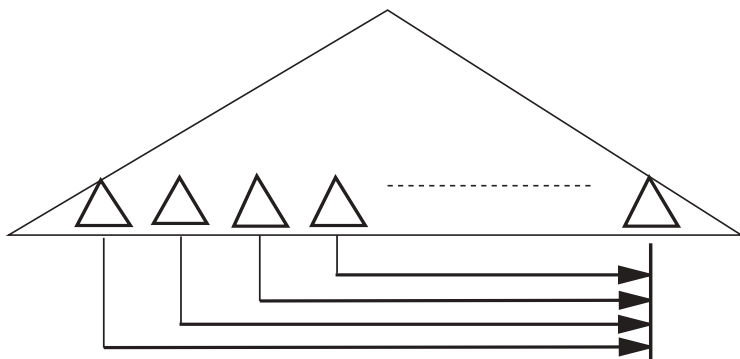


Fig. 10. The subtrees on the left will scan to the rightmost position to avoid memorizing the states for successor subtrees to use. The subtrees' scores and their respective positions will be propagated to their parent nodes so that bigger subtrees can move to the right.

We significantly reduced the high cost of computing recurrences by modifying the data storing strategy of the Walking Tree slightly. The new Walking Tree stores P , but not P 's inverse (Figure 9). Also, the new tree scans both T and T 's inverse. This new tree reduces the space and runtime to $\Theta(|L||T||\Sigma|^{|L|})$, i.e., reducing both runtime and space by a factor of $|\Sigma|^{|L|}$.

3.3 More Space Reduction

The $\Theta(|L||T||\Sigma|^{|L|})$ space cost can be reduced further by moving all trailer subtrees to scan the text string until they reach the position of the leading subtree. After all other subtrees scan to the leading one's position, we need only $\Theta(|L||\Sigma|^{|L|})$ space for recurrences because now all subtrees are at the leading subtree position (Figure 10). To achieve this space reduction, we have to move

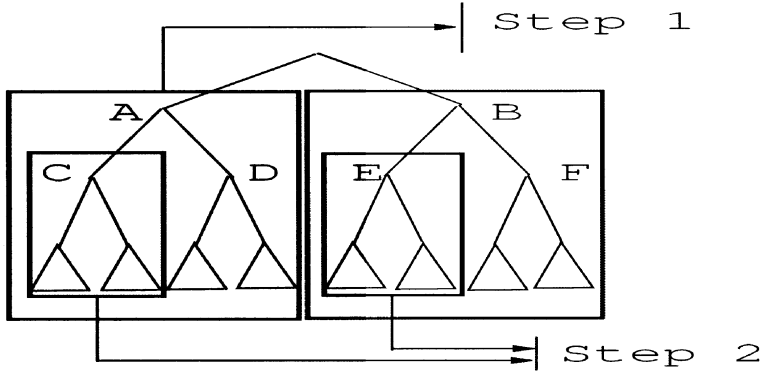


Fig. 11. First, we move the root's left subtree A to the position of the root's right subtree B, as indicated by Step 1. Then, we move A's left subtree and B's left subtree to the positions of A's and B's right subtrees, as indicated by Step 2. These recursive steps continue until the subtree we move has only $|L|$ leaves. When subtrees move, their scores and positions of the scores have to be stored. However, the space needed to store the scores and positions is only $\Theta(|P| \log |P|)$.

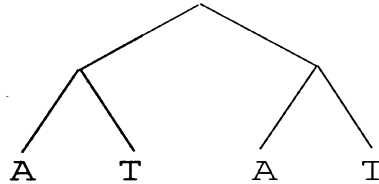


Fig. 12. We use the AT subtree to scan the entire text to generate scores and their respective positions. The score and position for the 4 element tree is computed using these stored results.

the subtrees in a recursive divide-and-conquer fashion. First, we move the root's left subtree A to the position of the root's right subtree B. Then, we move the left subtree of A, say C, and the left subtree of B, say E, to the position of the right subtrees of A and B. We do this recursively until the subtree we move has only $|L|$ leaves (Figure 11). This method reduces space usage by a factor of $\Theta(|T|)$.

4 Performance Characteristics

For the sequential version, the fast method runs in $\Theta(|L| |T| |\Sigma|^{|L|} + |P| |T| / |L|)$ time using $\Theta(|L| |\Sigma|^{|L|} + |P| \log |P|)$ space. If $|L|$ is chosen to be $\log_2 |P|/4$, we have the sub-quadratic sequential runtime $\Theta(|P| |T| / \log |P|)$. The parallel version runs in $\Theta((|L| |T| |\Sigma|^{|L|})/k + |P| |T| / (n|L|))$ time with n processors (where $k \leq n$ and $k \leq |\Sigma|^{|L|}$) using $\Theta(|L| |\Sigma|^{|L|} + |P| \log |P|)$ space. Since $(|L| |\Sigma|^{|L|})/k$ is much smaller than $(|P|/n|L|)$ in most cases, the speedup is $|L|$.

New Method's Speed / Improved Method's

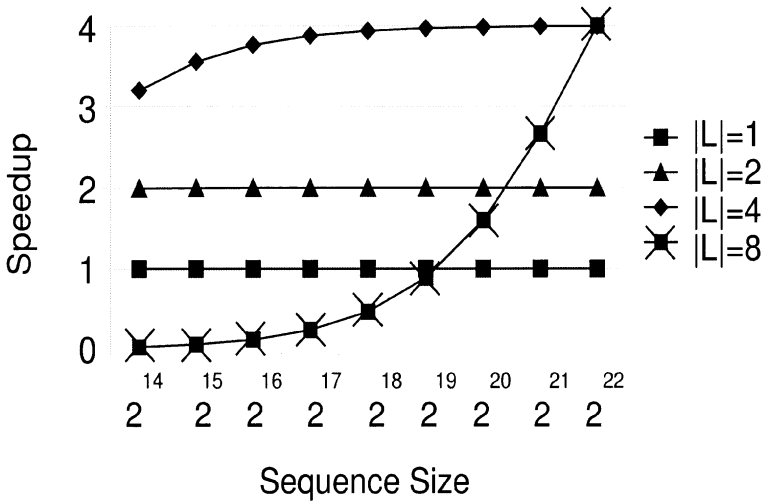


Fig. 13. For large sequence size, it is not difficult for the new method to get a speedup around 4, i.e., 400% in performance. To get a speedup over 4, the sequence size has to be greater than 2^{22} (about 4 million).

5 An Example

Here is an example to illustrate the recurrence-reduced Walking Tree Method. First, we scan the entire text string with a 2-leaf walking tree with leaves “AT” to generate scores at each scan position. Then, we use the scores and corresponding text positions generated by the 2-leaf Walking Tree for the 4-leaf Walking Tree. When the 4-leaf Walking Tree scans the text, it will look up the scores for its two 2-leaf subtrees at each scan position rather than recompute the scores (Figure 12).

6 Experiments

We compared the parallel fast method of of this paper to the parallel method B of the paper [7]. Both are coded in C and compiled using gcc version 2.7.2.3. Parallelization was provided by MPICH (version 1.1.0) [18]. The programs were run on cluster of Pentium III 850 MHz processors connected by a 100Mbps switch. Each processor ran Redhat Linux 6.0 and had sufficient local memory to avoid thrashing. We tested the programs on 33 processors and used randomly generated test strings of 2^{14} through 2^{22} characters. To be fair in the test, the ratio of the new method’s speed over the old method’s is normalized to 1 when $|L| = 1$, as shown in Figure 13. The speedup is defined as the new method’s speed divided by the old method’s, i.e. (the old method’s normalized runtime) / (the new method’s). As expected from Figure 13, the new method works very

well when $|L| = 2$ and $|L| = 4$, i.e., when the recurrence L -leaf subtree has only 2 or 4 leaves. We can easily get a speedup around 4, i.e., 400% in performance. To get a speedup over 4, the sequence size has to be larger than 2^{22} (about 4 million).

7 Conclusion

The walking tree technique provides methods to find biologically reasonable string alignment. Recent advances in genomics have depended on identification of genes and promotor regions on the basis of local information. While whole gene comparison are desired, they seem to be beyond the ability of current algorithms whose time and/or space complexity makes alignments of a hundred kilo bases impractical. Our continuing improvements of Walking Tree methods shows that a very high degree of parallelization is available, and can be exploited even on clusters of simple processors.

We have demonstrated that one million base pair sequences can be aligned within of a few hours on a cluster of 33 processors. Our refinements make alignments of ten mega bases strings practical within a week. Alignment of whole Eukaryote genomes will help to solve the too few genes for too many proteins paradox.

References

1. A. Caprara. Sorting by reversals is difficult. RECOMB 1997, pp75–83. ACM Press, New York.
2. Eugene Y. Chan. Molecular Motors. US Patent No.: US 6,210,896 B1, Apr. 3, 2001.
3. Eugene Y. Chan, Methods and Products for Analyzing Polymers. US Patent No.: US 6,355,420 B1, Mar. 12, 2002.
4. P. Cull and J. Holloway. Divide and Conquer Approximate String Matching, When Dynamic Programming is not Powerful Enough. Technical Report 92-20-06, Computer Science Department, Oregon State University, 1992.
5. P. Cull and J. Holloway. Aligning genomes with inversion and swaps. Second International Conference on Intelligent Systems for Molecular Biology. Proceedings of ISBM'94, AAAI Press, Menlo Park CA 1994, p. 195–202.
6. P. Cull, J. Holloway and J. Cavener, Walking Tree Heuristics for Biological String Alignment, Gene Location, and Phylogenies. CASYS'98, Computing Anticipatory Systems (D. M. Dubois, editor), American Institute of Physics, Woodbury, New York, pp201–215, 1999.
7. Paul Cull and Tai Hsu. Improved Parallel and Sequential Walking Tree Algorithms for Biological String Alignments. Supercomputing 99 Conference, 1999.
8. Paul Cull and Tai Hsu. Gene Verification and Discovery by Walking Tree Method. Pacific Symposium on Biocomputing 2001, World Scientific, Singapore, 2001, pp287–298.
9. K. M. Devos, M. D. Atkinson, C. N. Chinoy, H. A. Francis, R. L. Harcourt, R. M. D. Koebner, C. J. Liu, P. Masojc, D. X. Xie, and M. D. Gale. Chromosomal rearrangements in the rye genome relative to that of wheat. Theoretical and Applied Genetics 85:673–680, 1993.

10. S. Hannenhalli. Polynomial algorithm for computing translocation distance between genomes. *Combinatorial Pattern Matching*, pp162–176, 1995.
11. S. Hannenhalli and P. Pevzner. Transforming cabbage into turnip: polynomial algorithm for sorting signed permutations by reversals. *Proceedings of the 27th ACM Symposium on the Theory of Computing*, pp178–189, 1995.
12. M. Schöniger and M. S. Waterman, A local algorithm for DNA sequence alignment with inversions. *Bulletin of Mathematical Biology*, 54:521–536, 1992.
13. R. S. Stephens, S. Kalman, C. Lammel and colleagues. Genome Sequence of an Obligate Intracellular Pathogen of Humans: *Chlamydia Trachomatis*. *Science* 282:754–759, 1998.
14. J. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology*. PWS Publishing, Boston, MA, 1996.
15. Waltham Rudolf Gilmanshin and Eugene Y. Chan. *Methods of Analyzing Polymers using Spatial Network of Fluorophores and Fluorescence Resonance Energy Transfer*, US Patent No.: US 6,263,286 B1, Jul. 17, 2001.
16. C. M. Fraser, S. Casjens, W. M. Huang, et al. Genomic sequence of a Lyme disease spirochaete, *Borrelia burgdorferi*. *Nature* 1997 Dec, 390(6660):580–586.
17. V. L. Arlazarov, E. A. Dinic, M. A. Kronrod, and I. A. Faradzev. On economic construction of the transitive closure of a directed graph. *Dokl. Acad. Nauk SSSR*, 194:487–88, 1970.
18. W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A high-performance, portable implementation of the MPI message passing interface standard. *Parallel Computing*, vol. 22, no. 6, p.789–828, September, 1996.
19. M. Python. *Just The Words*. Methuen, London, 1989, Vol. 2, p.338–339.

Towards Some Computational Problems Arising in Biological Modeling^{*}

Virginia Giorno¹, Amelia G. Nobile¹, Enrica Pirozzi², and Luigi M. Ricciardi²

¹ Università di Salerno,

Dipartimento di Matematica e Informatica,

84041 Baronissi (SA), Italy

{giorno,nobile}@unisa.it

² Università di Napoli Federico II,

Dipartimento di Matematica e Applicazioni,

80126 Napoli, Italy

{enrica.pirozzi,luigi.ricciardi}@unina.it

Abstract. Time-nonhomogeneous diffusion processes confined by a time dependent reflecting boundary are investigated to obtain a system of integral equations concerning the transition pdf in the presence of the reflecting boundary. For Gauss-Markov processes restricted by particular time dependent reflecting boundaries a closed form solution of the transition pdf is derived. Furthermore, the first passage time problem through time-dependent thresholds is analyzed and a nonsingular second-kind Volterra integral equation for the first passage time pdf is obtained.

1 Introduction and Background

One dimensional diffusion processes have been increasingly used as approximations to the description of the evolution of intrinsically discrete systems such as natural populations (see, for instance, [9], [10], [11], [13]) and neuronal systems (see, for instance, [1], [3], [5], [8], [13]). In studies of population dynamics with immigration effects, the number of individuals is bound to take non negative values, so that a reflection condition at zero is customarily imposed. Instead, in neuronal modeling the membrane potential evolution can be described by focusing the attention on stochastic processes confined by a reflecting boundary that can be looked at as the neuronal reversal hyperpolarization potential. In both types of instances, first-passage-time densities are to be invoked to describe events such as extinction of populations (corresponding to the attainment of close-to-zero sizes) and neuronal firings, that originates when a suitable threshold value is reached by the modeled time-course of the membrane potential. In order to be able to obtain such first-passage-time densities, the knowledge of

^{*} Work performed within a joint cooperation agreement between Japan Science and Technology Corporation (JST) and Università di Napoli Federico II, under partial support by INdAM (G.N.C.S).

the free transition probability density function (pdf) is not sufficient. Neither it is sufficient to obtain transition densities in the presence of preassigned time reflecting boundaries that play an essential role in modeling formulations.

Due to its relevance for a quantitative description of biological instances such as those mentioned above, here we shall concentrate on the determination of a system of integral equations that allows one to express the transition pdf in the presence of a reflecting boundary for a time-nonhomogeneous diffusion process in terms of the free transition pdf and of the probability current of the considered process. It is essential to emphasize that for both Gauss-Markov processes and for diffusion processes the probability current can be expressed in terms of the free transition pdf of the considered process. Since the free probability density is known in closed form for Gauss-Markov processes, use of it will be made to determine transition pdf's in the presence of particular reflecting boundaries, in view of their potential interest for various applications.

Let $\{X(t), t \in T\}$ be a time-nonhomogeneous diffusion process defined over the interval $I \equiv (r_1, r_2)$ and let $A_1(x, t)$ and $A_2(x, t)$ be the drift and infinitesimal variance of $X(t)$, respectively. For all $\tau < t$ and $x, y \in I$ we consider the following functions:

$$\begin{aligned} F(x, t|y, \tau) &= P\{X(t) < x | X(\tau) = y\} \quad (\text{transition distribution}), \\ f(x, t|y, \tau) &= \frac{\partial}{\partial x} F(x, t|y, \tau) \quad (\text{transition pdf}). \end{aligned}$$

The transition pdf $f(x, t|y, \tau)$ of $X(t)$ can be obtained as the solution of the Fokker-Planck equation

$$\frac{\partial}{\partial t} f(x, t|y, \tau) = -\frac{\partial}{\partial x} \left[A_1(x, t) f(x, t|y, \tau) \right] + \frac{1}{2} \frac{\partial^2}{\partial x^2} \left[A_2(x, t) f(x, t|y, \tau) \right], \quad (1)$$

with the delta initial condition

$$\lim_{t \downarrow \tau} f(x, t|y, \tau) = \delta(x - y). \quad (2)$$

Further, the transition distribution is solution of the following equation

$$\begin{aligned} \frac{\partial}{\partial t} F(x, t|y, \tau) &= -\left[A_1(x, t) - \frac{1}{2} \frac{\partial A_2(x, t)}{\partial x} \right] \frac{\partial}{\partial x} F(x, t|y, \tau) \\ &\quad + \frac{1}{2} A_2(x, t) \frac{\partial^2}{\partial x^2} F(x, t|y, \tau), \end{aligned} \quad (3)$$

with the initial and boundary conditions

$$\lim_{t \downarrow \tau} F(x, t|y, \tau) = \begin{cases} 0, & x < y \\ 1/2, & x = y \\ 1, & x > y, \end{cases} \quad (4)$$

$$\lim_{x \downarrow r_1} F(x, t|y, \tau) = 0, \quad \lim_{x \uparrow r_2} F(x, t|y, \tau) = 1. \quad (5)$$

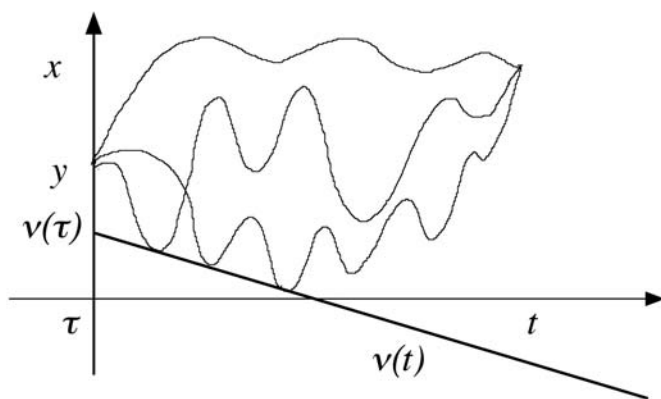


Fig. 1. Sample paths originating in y at the time τ and ending in x at time t , for the process $X(t)$ confined at the interval $[\nu(t), r_2]$ by the reflecting boundary $\nu(t)$

Let $\nu(t)$ be a function of class $C^1(T)$ taking values in I . We define the following functions:

$$j(x, t|y, \tau) = A_1(x, t) f(x, t|y, \tau) - \frac{1}{2} \frac{\partial}{\partial x} \left\{ A_2(x, t) f(x, t|y, \tau) \right\}, \quad (6)$$

$$J[\nu(t), t|y, \tau] = -\frac{d}{dt} F[\nu(t), t|y, \tau] = j[\nu(t), t|y, \tau] - \nu'(t) f[\nu(t), t|y, \tau], \quad (7)$$

where the prime denoting derivative with respect to the argument. We note that (6) is the definition of the probability current.

In the following, we shall assume that the process $X(t)$ is restricted to $[\nu(t), r_2]$, with a reflecting condition imposed on the boundary $\nu(t)$. For all $t > \tau$, the sample paths of this process are confined in $[\nu(t), r_2]$ (cf. Fig. 1).

The transition pdf $f_r(x, t|y, \tau)$ of $X(t)$ in the presence of the reflecting boundary $\nu(t)$ can be obtained as the solution of the Fokker-Planck equation (1) with the delta initial condition (2) and the additional requirement that the total probability mass is conserved in $[\nu(t), r_2]$, i.e.

$$\int_{\nu(t)}^{r_2} f_r(x, t|y, \tau) dx = 1 \quad [y \geq \nu(\tau)]. \quad (8)$$

The transition distribution $F_r(x, t|y, \tau)$ of $X(t)$ in the presence of the reflecting boundary $\nu(t)$ is solution of (3) with the boundary condition

$$\lim_{x \uparrow r_2} F_r(x, t|y, \tau) = \int_{\nu(t)}^{r_2} f_r(x, t|y, \tau) dx = 1, \quad y \geq \nu(\tau) \quad (9)$$

and the initial condition (4), for $y > \nu(\tau)$ and

$$\lim_{t \downarrow \tau} F_r[x, t|\nu(\tau), \tau] = \begin{cases} 0, & x = \nu(\tau) \\ 1, & x > \nu(\tau), \end{cases} \quad (10)$$

if $y = \nu(\tau)$.

In Section 2 we shall prove that the knowledge of the free transition pdf f permits us to obtain a system of integral equations that can be used to determine a numerical evaluation of the the transition pdf f_r .

In Section 3, we shall restrict our attention to Gauss-Markov processes. For these processes and for a particular class of reflecting boundaries, we shall determine f_r in closed form and study the first passage time problem.

2 Mathematical Results

Lemma 1. *Let $X(t)$ be a time-nonhomogeneous diffusion process restricted to the interval $[\nu(t), r_2)$ with $\nu(t)$ a reflecting boundary and let*

$$J_n[\nu(t), t|y, \tau] = \begin{cases} J[\nu(t), t|y, \tau], & n = 1 \\ \int_{\tau}^t J[\nu(\vartheta), \vartheta|y, \tau] J_{n-1}[\nu(t), t|\nu(\vartheta), \vartheta] d\vartheta, & n \geq 2 \end{cases} \quad (11)$$

for $y \geq \nu(\tau)$, with J defined as in (7). Then, for $n \geq 2$ and $y \geq \nu(\tau)$, one has:

$$J_n[\nu(t), t|y, \tau] = \int_{\tau}^t J_{n-1}[\nu(\vartheta), \vartheta|y, \tau] J[\nu(t), t|\nu(\vartheta), \vartheta] d\vartheta. \quad (12)$$

Proof. Due to the first of (11), it is immediately seen that relations (11) and (12) are equivalent for $n = 2$. Let $y = \nu(\tau)$. We now proceed by induction and prove that if (12) holds for an arbitrarily fixed n , it also holds for $n + 1$. Indeed, from (11), by induction and making use of Fubini's theorem, one obtains:

$$\begin{aligned} J_{n+1}[\nu(t), t|\nu(\tau), \tau] &= \int_{\tau}^t J[\nu(\vartheta), \vartheta|\nu(\tau), \tau] J_n[\nu(t), t|\nu(\vartheta), \vartheta] d\vartheta \\ &= \int_{\tau}^t J[\nu(\vartheta), \vartheta|\nu(\tau), \tau] d\vartheta \int_{\vartheta}^t J_{n-1}[\nu(u), u|\nu(\vartheta), \vartheta] J[\nu(t), t|\nu(u), u] du \\ &= \int_{\tau}^t J[\nu(t), t|\nu(u), u] du \int_{\tau}^u J[\nu(\vartheta), \vartheta|\nu(\tau), \tau] J_{n-1}[\nu(u), u|\nu(\vartheta), \vartheta] d\vartheta \\ &= \int_{\tau}^t J[\nu(t), t|\nu(u), u] J_n[\nu(u), u|\nu(\tau), \tau] du, \end{aligned}$$

where the last equality follows from (11). Hence, (12) holds for all for $n \geq 2$ if $y = \nu(\tau)$. Now, we prove (12) for $y > \nu(\tau)$. From (11), by use of (12) for $y = \nu(\tau)$ and of Fubini's theorem, we obtain:

$$\begin{aligned} J_n[\nu(t), t|y, \tau] &= \int_{\tau}^t J[\nu(\vartheta), \vartheta|y, \tau] J_{n-1}[\nu(t), t|\nu(\vartheta), \vartheta] d\vartheta \\ &= \int_{\tau}^t J[\nu(\vartheta), \vartheta|y, \tau] d\vartheta \int_{\vartheta}^t J_{n-2}[\nu(u), u|\nu(\vartheta), \vartheta] J[\nu(t), t|\nu(u), u] du \\ &= \int_{\tau}^t J[\nu(t), t|\nu(u), u] du \int_{\tau}^u J[\nu(\vartheta), \vartheta|y, \tau] J_{n-2}[\nu(u), u|\nu(\vartheta), \vartheta] d\vartheta \end{aligned}$$

$$= \int_{\tau}^t J[\nu(t), t|\nu(u), u] J_{n-1}[\nu(u), u|y, \tau] du,$$

where again use of (11) has been made. This completes the proof.

Theorem 1. *Let $X(t)$ be a time-nonhomogeneous diffusion process restricted into the interval $[\nu(t), r_2)$ with $\nu(t)$ reflecting boundary. Then*

$$\begin{aligned} 1 - F_r(x, t|y, \tau) &= 1 - F(x, t|y, \tau) \\ &+ \sum_{n=1}^{+\infty} (-2)^n \int_{\tau}^t J_n[\nu(\vartheta), \vartheta|y, \tau] \{1 - F[x, t|\nu(\vartheta), \vartheta]\} d\vartheta \\ &\quad [x \geq \nu(t), y > \nu(\tau)] \end{aligned} \quad (13)$$

$$\begin{aligned} 1 - F_r[x, t|\nu(\tau), \tau] &= 2 \{1 - F[x, t|\nu(\tau), \tau]\} \\ &+ 2 \sum_{n=1}^{+\infty} (-2)^n \int_{\tau}^t J_n[\nu(\vartheta), \vartheta|\nu(\tau), \tau] \{1 - F[x, t|\nu(\vartheta), \vartheta]\} d\vartheta \\ &\quad [x \geq \nu(t)], \end{aligned} \quad (14)$$

with J_n defined in (11).

Proof. For $x \geq \nu(t)$ and $y > \nu(\tau)$, let

$$F^*(x, t|y, \tau) = F(x, t|y, \tau) - \sum_{n=1}^{+\infty} (-2)^n \int_{\tau}^t J_n[\nu(\vartheta), \vartheta|y, \tau] \{1 - F[x, t|\nu(\vartheta), \vartheta]\} d\vartheta.$$

We now prove that $F^*(x, t|y, \tau)$ satisfies (3) for $x > \nu(t)$ and $y > \nu(\tau)$. Since

$$\lim_{\vartheta \uparrow t} F[x, t|\nu(\vartheta), \vartheta] = 1, \quad x > \nu(t),$$

one has:

$$\begin{aligned} \frac{\partial}{\partial t} F^*(x, t|y, \tau) &= \frac{\partial}{\partial t} F(x, t|y, \tau) \\ &- \sum_{n=1}^{+\infty} (-2)^n \int_{\tau}^t J_n[\nu(\vartheta), \vartheta|y, \tau] \frac{\partial}{\partial t} F[x, t|\nu(\vartheta), \vartheta] d\vartheta. \end{aligned} \quad (15)$$

Furthermore, for $i = 1, 2$, there results:

$$\begin{aligned} \frac{\partial^i}{\partial x^i} F^*(x, t|y, \tau) &= \frac{\partial^i}{\partial x^i} F(x, t|y, \tau) \\ &+ \sum_{n=1}^{+\infty} (-2)^n \int_{\tau}^t J_n[\nu(\vartheta), \vartheta|y, \tau] \frac{\partial^i}{\partial x^i} F[x, t|\nu(\vartheta), \vartheta] d\vartheta. \end{aligned} \quad (16)$$

Recalling that the free transition distribution F satisfies (3), it is then easy to prove that also $F^*(x, t|y, \tau)$ satisfies (3). Furthermore, since for $y > \nu(\tau)$ and $x \geq \nu(t)$ one has:

$$\lim_{t \downarrow \tau} F^*(x, t|y, \tau) \equiv \lim_{t \downarrow \tau} F(x, t|y, \tau)$$

and

$$\lim_{x \uparrow r_2} F^*(x, t|y, \tau) = 1,$$

it follows that $F^*(x, t|y, \tau)$ satisfies (4) and (9). Hence, for $x \geq \nu(t)$ and $y > \nu(\tau)$, $F_r(x, t|y, \tau) \equiv F^*(x, t|y, \tau)$ satisfies (13). Similarly, for $x \geq \nu(t)$ and $y = \nu(\tau)$, setting

$$\begin{aligned} F^*[x, t|\nu(\tau), \tau] &= -1 + 2 F[x, t|\nu(\tau), \tau] \\ &\quad - 2 \sum_{n=1}^{+\infty} (-2)^n \int_{\tau}^t J_n[\nu(\vartheta), \vartheta|\nu(\tau), \tau] \{1 - F[x, t|\nu(\vartheta), \vartheta]\} d\vartheta, \end{aligned}$$

one can easily prove that $F^*[x, t|\nu(\tau), \tau]$ satisfies (3) for $x > \nu(t)$ and (9) for $x \geq \nu(t)$. Further, since

$$\lim_{t \downarrow \tau} F^*[x, t|\nu(\tau), \tau] = -1 + 2 \lim_{t \downarrow \tau} F[x, t|\nu(\tau), \tau] = \begin{cases} 0, & x = \nu(\tau) \\ 1, & x > \nu(\tau), \end{cases}$$

$F^*[x, t|\nu(\tau), \tau]$ satisfies (10). Hence, for $x \geq \nu(t)$, we conclude that $F_r[x, t|\nu(\tau), \tau] \equiv F^*[x, t|\nu(\tau), \tau]$ satisfies (14).

Theorem 2. *Under the assumption of Theorem 1, we have:*

$$\begin{aligned} 1 - F_r(x, t|y, \tau) &= 1 - F(x, t|y, \tau) - \int_{\tau}^t J[\nu(\vartheta), \vartheta|y, \tau] \{1 - F_r[x, t|\nu(\vartheta), \vartheta]\} d\vartheta \\ &\quad [x \geq \nu(t), y > \nu(\tau)] \end{aligned} \quad (17)$$

$$\begin{aligned} 1 - F_r[x, t|\nu(\tau), \tau] &= 2 \{1 - F[x, t|\nu(\tau), \tau]\} \\ &\quad - 2 \int_{\tau}^t J[\nu(\vartheta), \vartheta|\nu(\tau), \tau] \{1 - F_r[x, t|\nu(\vartheta), \vartheta]\} d\vartheta \\ &\quad [x \geq \nu(t)], \end{aligned} \quad (18)$$

with J defined in (7).

Proof. Let $x \geq \nu(t)$ and $y > \nu(t)$. From (13), recalling (11) and Fubini's theorem, we have:

$$\begin{aligned} 1 - F_r(x, t|y, \tau) &= 1 - F(x, t|y, \tau) - 2 \int_{\tau}^t J[\nu(\theta), \theta|y, \tau] \{1 - F[x, t|\nu(\vartheta), \vartheta]\} d\vartheta \\ &\quad + \sum_{n=1}^{+\infty} (-2)^{n+1} \int_{\tau}^t du \{1 - F[x, t|\nu(u), u]\} \int_{\tau}^u d\vartheta J[\nu(\vartheta), \vartheta|y, \tau] J_n[\nu(u), u|\nu(\vartheta), \vartheta] \end{aligned}$$

$$\begin{aligned}
&= 1 - F(x, t|y, \tau) - \int_{\tau}^t d\vartheta J[\nu(\vartheta), \vartheta|y, \tau] \left[2 \{1 - F[x, t|\nu(\vartheta), \vartheta]\} \right. \\
&\quad \left. + 2 \sum_{n=1}^{+\infty} (-2)^n \int_{\vartheta}^t du J_n[\nu(u), u|\nu(\vartheta), \vartheta] \{1 - F[x, t|\nu(u), u]\} \right] \\
&= 1 - F(x, t|y, \tau) - \int_{\tau}^t J[\nu(\vartheta), \vartheta|y, \tau] \{1 - F[x, t|\nu(\vartheta), \vartheta]\} d\vartheta.
\end{aligned}$$

where the last equality follows from (14). Therefore (17) holds. Similarly, for $x \geq \nu(t)$ and $y = \nu(\tau)$, Equation (18) can be obtained from (14).

Theorem 3. *Under the assumption of Theorem 1 one has:*

$$\begin{aligned}
f_r(x, t|y, \tau) &= f(x, t|y, \tau) - \int_{\tau}^t J[\nu(\vartheta), \vartheta|y, \tau] f_r[x, t|\nu(\vartheta), \vartheta] d\vartheta \\
&\quad [x \geq \nu(t), y > \nu(\tau)] \quad (19)
\end{aligned}$$

$$\begin{aligned}
f_r[x, t|\nu(\tau), \tau] &= 2 f[x, t|\nu(\tau), \tau] - 2 \int_{\tau}^t J[\nu(\vartheta), \vartheta|\nu(\tau), \tau] f_r[x, t|\nu(\vartheta), \vartheta] d\vartheta \\
&\quad [x \geq \nu(t)], \quad (20)
\end{aligned}$$

with J defined in (7).

Proof. Equations (19) and (20) immediately follow by differentiating both sides of (17) and (18) with respect to x .

We note that results of Theorem 3 are in accordance with the results previously obtained for time homogeneous diffusion processes and time constant reflecting boundaries (cf., for instance, [7] and [12]).

Equation (20) shows that the function $f_r[x, t|\nu(\tau), \tau]$ is solution of a second-kind Volterra integral equation. We emphasize the utility, from a numerical point of view, of the equations system (19) and (20). Indeed, the knowledge of the free transition pdf $f(x, t|y, \tau)$ of the process $X(t)$ is sufficient to obtain a numerical solution of Equation (19) via standard techniques. Such numerical solution gives $f_r[x, t|\nu(\tau), \tau]$ for all $t \geq \tau$ that can be used to solve numerically Equation (19), thus obtaining a numerical evaluation of the function $f_r(x, t|y, \tau)$.

3 Gauss-Markov Processes

We now restrict our attention to Gauss-Markov processes.

Let $m(t), h_1(t), h_2(t)$ be $C^1(T)$ -class functions such that $r(t) = h_1(t)/h_2(t)$ is monotonically increasing function. If $\{W(t), t \geq 0\}$ denote a standard Wiener process, then

$$X(t) = m(t) + h_2(t) W[r(t)] \quad (21)$$

is a non-singular Gauss-Markov process with mean $m(t)$ and covariance $c(s, t) = h_1(s) h_2(t)$ for $s \leq t$. The free transition pdf $f(x, t|y, \tau)$ of $X(t)$ is a normal density characterized respectively by mean and variance:

$$E[X(t)|X(\tau) = y] = m(t) + \frac{h_2(t)}{h_2(\tau)} [y - m(\tau)], \quad (22)$$

$$Var[X(t)|X(\tau) = y] = h_2(t) \left[h_1(t) - \frac{h_2(t)}{h_2(\tau)} h_1(\tau) \right], \quad (23)$$

for $t, \tau \in T$ and $\tau < t$. Furthermore, $f(x, t|y, \tau)$ satisfies the Fokker-Planck equation (1) and the associated initial delta condition (2), with $A_1(x, t)$ and $A_2(x, t) \equiv A_2(t)$ given by

$$A_1(x, t) = m'(t) + [x - m(t)] \frac{h_2'(t)}{h_2(t)}, \quad A_2(t) = h_2^2(t) r'(t), \quad (24)$$

the prime denoting derivative with respect to the argument. Since the free transition pdf $f(x, t|y, \tau)$ is a normal density with mean and variance respectively given in (22) and (23), from (7) one has:

$$\begin{aligned} J[\nu(t), t|y, \tau] = f[\nu(t), t|y, \tau] & \left\{ -[\nu'(t) - m'(t)] + [\nu(t) - m(t)] \frac{h_2'(t)}{h_2(t)} \right. \\ & \left. + \frac{h_2(\tau) h_2(t) r'(t)}{2[h_1(t) h_2(\tau) - h_1(\tau) h_2(t)]} \left(\nu(t) - m(t) - \frac{h_2(t)}{h_2(\tau)} [y - m(\tau)] \right) \right\}. \end{aligned} \quad (25)$$

Due to the circumstance that for a Gauss-Markov process the functions f and J are known, Equations (19) and (20) can be solved via a numerical procedures to obtain the function $f_r(x, t|y, \tau)$ in the presence of any reflecting boundary $\nu(t)$ of $C^1(T)$ -class. However, choosing the boundary $\nu(t)$ in a suitable way, it is possible to determine the transition pdf in the presence of the reflecting boundary $\nu(t)$ in closed form as the following theorem shows.

Theorem 4. *Let $X(t)$ be a non singular Gauss-Markov process characterized by mean $m(t)$ and covariance $c(s, t) = h_1(s) h_2(t)$ for $s \leq t$, confined at the interval $[\nu(t), +\infty)$ with*

$$\nu(t) = m(t) + a h_1(t) + b h_2(t), \quad a, b \in \mathbf{R} \quad (26)$$

a reflecting boundary. Then,

$$\begin{aligned} f_r(x, t|y, \tau) = f(x, t|y, \tau) - \frac{\partial}{\partial x} & \left\{ \varphi(x, t) F[\psi(x, t), t|y, \tau] \right\} \\ & [x \geq \nu(t), y \geq \nu(\tau)], \end{aligned} \quad (27)$$

where $f(x, t|y, \tau)$ is a normal density with mean (22) and variance (23), $F[\psi(x, t), t|y, \tau]$ is the corresponding probability distribution and

$$\varphi(x, t) = \exp\left(-\frac{2a}{h_2(t)} [x - \nu(t)]\right), \quad \psi(x, t) = 2\nu(t) - x. \quad (28)$$

Proof. Let

$$f_1(x, t|y, \tau) = f(x, t|y, \tau) - \frac{\partial}{\partial x} \left\{ \varphi(x, t) F[\psi(x, t), t|y, \tau] \right\}. \quad (29)$$

be the right-hand side of Equation (27). We prove that the function f_1 satisfies (1), (2) and (8) with $A_1(x, t)$ and $A_2(x)$ given in (24). To this purpose we note that, since f is a normal density, from (28), one has:

$$\varphi(x, t) f[\psi(x, t), t|y, \tau] = \varphi[\psi^{-1}(y, \tau), \tau] f(x, t|\psi^{-1}(y, \tau), \tau),$$

so that

$$\begin{aligned} f_1(x, t|y, \tau) &= f(x, t|y, \tau) + \varphi[\psi^{-1}(y, \tau), \tau] f(x, t|\psi^{-1}(y, \tau), \tau) \\ &\quad - \frac{\partial \varphi(x, t)}{\partial x} F[\psi(x, t), t|y, \tau]. \end{aligned}$$

Since the functions $f(x, t|y, \tau)$, $f(x, t|\psi^{-1}(y, \tau), \tau)$ and

$$f_2(x, t|y, \tau) = \frac{\partial \varphi(x, t)}{\partial x} F[\psi(x, t), t|y, \tau] = -\frac{2a}{h_2(t)} \varphi(x, t) F[\psi(x, t), t|y, \tau]$$

satisfy (1) for $x > \nu(t)$ and $y \geq \nu(\tau)$, also the right-hand side of (27) satisfies (1) for $x > \nu(t)$ and $y \geq \nu(\tau)$. Furthermore, (29) satisfies conditions (2) and (8). Hence, the left-hand side of (29) identifies with $f_r(x, t|y, \tau)$, i.e. (27) holds.

Example 1. Let $\{X(t), t \in [0, +\infty)\}$ be the Gauss-Markov process with

$$m(t) = t e^{-\mu t}, \quad (30)$$

$$c(s, t) = \frac{e^{-\mu(t-s)}}{2\mu} \quad (0 \leq s \leq t < +\infty, \mu > 0). \quad (31)$$

In this case one has:

$$h_1(t) = \frac{e^{\mu t}}{2\mu}, \quad h_2(t) = e^{-\mu t}, \quad r(t) = \frac{h_1(t)}{h_2(t)} = \frac{e^{2\mu t}}{2\mu}.$$

Hence, the free transition pdf $f(x, t|y, \tau)$ is a normal density characterized respectively by mean and variance:

$$E[X(t)|X(\tau) = y] = t e^{-\mu t} + e^{-\mu(t-\tau)} [y - \tau e^{-\mu \tau}] \quad (32)$$

$$Var[X(t)|X(\tau) = y] = \frac{1}{2\mu} [1 - e^{-2\mu(t-\tau)}]. \quad (33)$$

Furthermore, $f(x, t|y, \tau)$ satisfies the Fokker-Planck equation (1) with

$$A_1(x, t) = -\mu x + e^{-\mu t}, \quad A_2(x, t) = 1.$$

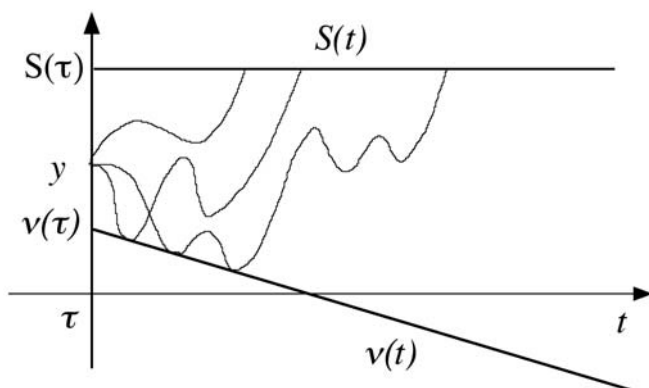


Fig. 2. Same sample paths related to FPT $\mathcal{T}_y$ from $X(\tau) = y \geq \nu(\tau)$ to $S(t)$ for the process $X(t)$ restricted to the interval $[\nu(t), +\infty)$ with $\nu(t)$ reflecting boundary

For Theorem 4, the transition pdf in the presence of the reflecting boundary

$$\nu(t) = m(t) + a h_1(t) + b h_2(t) \equiv (t + b) e^{-\mu t} + \frac{a}{2\mu} e^{\mu t},$$

with a and b real constants, is:

$$f_r(x, t|y, \tau) = f(x, t|y, \tau) - \frac{\partial}{\partial x} \left\{ \exp\left(-2a e^{\mu t} [x - \nu(t)]\right) F[2\nu(t) - x, t|y, \tau] \right\} \\ [x \geq \nu(t), y \geq \nu(\tau)],$$

where $f(x, t|y, \tau)$ is a normal density with mean (32) and variance (33) and F is the corresponding probability distribution.

For the Gauss-Markov process confined in the interval $[\nu(t), +\infty)$ with reflecting boundary $\nu(t)$ given in (26), the FPT problem to a boundary $S(t) > \nu(t)$ can be considered. To this purpose, let $S(t)$ be a $C^1(T)$ -class function, such that $S(t) > \nu(t)$ for all $t \in T$. For $y \geq \nu(\tau)$ we denote by

$$\mathcal{T}_y = \inf_{t \geq \tau} \{t : X(t) > S(t)\}, \quad X(\tau) = y < S(\tau), \quad \tau, t \in T \quad (34)$$

the random variable FPT of $X(t)$ from $X(\tau) = y \geq \nu(\tau)$ to the boundary $S(t)$ (cf. Fig. 2).

Further, let

$$g_r[S(t), t|y, \tau] = \frac{\partial}{\partial t} P(\mathcal{T}_y < t) \quad (35)$$

be the FPT pdf. It satisfies the following nonsingular second-kind Volterra integral equation (cf., for instance, [2], [4], [6])

$$g_r[S(t), t|y, \tau] = -2\Psi_r[S(t), t|y, \tau] + 2 \int_{\tau}^t g_r[S(\vartheta), \vartheta|y, \tau] \Psi_r[S(t), t|S(\vartheta), \vartheta] d\vartheta \\ [y < S(\tau)], \quad (36)$$

where

$$\Psi_r[S(t), t|y, \tau] = \left\{ \frac{S'(t) - m'(t)}{2} - \frac{S(t) - m(t)}{2} \frac{h'_1(t)h_2(\tau) - h'_2(t)h_1(\tau)}{h_1(t)h_2(\tau) - h_2(t)h_1(\tau)} - \frac{y - m(\tau)}{2} \frac{h'_2(t)h_1(t) - h_2(t)h'_1(t)}{h_1(t)h_2(\tau) - h_2(t)h_1(\tau)} \right\} f_r[S(t), t|y, \tau] \quad (37)$$

with $f_r[S(t), t|y, \tau]$ given by (27). We note that since the kernel of the (36) is not singular the equation can be solved through standard numerical procedures (cf., for instance, [4]).

In conclusion, we should point out that a systematic computational analysis has to be associated to the theoretical results outlined above, if specific models have to be tested and their robustness estimated. Such an analysis, requiring use of large scale vector or parallel computers, is currently the object of some of our current investigations.

References

1. Buonocore, A., Giorno, V., Nobile, A.G. and Ricciardi, L.M.: A neuronal modeling paradigm in the presence of refractoriness. *BioSystems*, Vol. 67 (2002) 35–43
2. Buonocore, A., Nobile, A.G. and Ricciardi, L.M.: A new integral equation for the evaluation of first-passage-time probability densities. *Adv. Appl. Prob.*, Vol. 19 (1987) 784–800
3. Di Crescenzo, A., Di Nardo, E., Nobile, A.G., Pirozzi, E. and Ricciardi, L.M.: On some computational results for single neurons' activity modeling. *BioSystems*, Vol. 58 (2000) 19–26
4. Di Nardo, E., Nobile, A.G., Pirozzi, E. and Ricciardi, L.M.: A computational approach to first-passage-time problems for Gauss-Markov processes. *Adv. Appl. Prob.*, Vol. 33 (2001) 453–482
5. Giorno, V., Lánský, P., Nobile, A.G. and Ricciardi, L.M.: Diffusion approximation and first-passage-time problem for a model neuron. III. A birth-and-death process approach. *Biol. Cybern.*, Vol. 58 (1988) 387–404.
6. Giorno, V., Nobile, A.G., Ricciardi, L.M. and Sato, S.: On the evaluation of first-passage-time probability densities via non-singular integral equation. *Adv. Appl. Prob.*, Vol. 21 (1989) 20–36.
7. Giorno, V., Nobile, A.G. and Ricciardi, L.M.: On the transition densities of diffusion processes with reflecting boundaries. In Trappé, R. (ed.): *Cybernetics and Systems '90. Proceedings of the Tenth Meeting on Cybernetics and Systems Research*, Vol. 1. World Scientific, Singapore New Jersey London Hong Kong (1990) 383–390.
8. Giorno, V., Nobile, A.G. and Ricciardi, L.M.: Single neuron's activity: on certain problems of modeling and interpretation. *BioSystems*, Vol. 40 (1997) 65–74
9. Nobile, A.G. and Ricciardi, L.M.: Growth with regulation in fluctuating environments. I. Alternative logistic-like diffusion models. *Cybernetics*, Vol. 49 (1984) 179–188
10. Nobile, A.G. and Ricciardi, L.M.: Growth with regulation in fluctuating environments. II. Intrinsic lower bounds to population size. *Biol. Cybernetics*, Vol. 50 (1984) 285–299

11. Ricciardi, L.M.: Diffusion Processes and Related Topics in Biology. Springer-Verlag, New York (1977)
12. Ricciardi, L.M. and Sacerdote, L.: On the probability densities of an Ornstein-Uhlenbeck process with a reflecting boundary. J. Appl. Prob., Vol. 24 (1987) 355–369
13. Ricciardi, L.M., Di Crescenzo, A., Giorno, V. and Nobile, A.G.: An outline of theoretical and algorithmic approaches to first passage time problems with applications to biological modeling. Math. Japonica, Vol. 50 No.2 (1999) 247–322

Single Point Algorithms in Genetic Linkage Analysis

Daniel Gudbjartsson¹, Jens A. Hansen², Anna Ingólfssdóttir¹,
Jacob Johnsen², and John Knudsen²

¹ DeCode Genetics, Reykjavik, Iceland
annai@decode.is

² Basic Research in Computer Science,
Centre of the Danish National Research Foundation,
Department of Computer Science, Aalborg University,
Fr. Bajersvej 7E, 9220 Aalborg Ø, Denmark

Abstract. In this paper we provide a mathematical model that describes genetic inheritance. In terms of our framework, we describe two commonly used algorithms that calculate the set of founder allele assignment that are compatible with some genotype information and a given inheritance pattern. These calculations cater for the first step in the multi point linkage analysis. The first algorithm we investigate is a part of the GENHUNTER software package developed at the Whitehead Institute at MIT and is developed by Kruglyak et al [8]; the second one occurs in the software package ALLEGRO provided by Gudbjartsson et al [5] at DeCode Genetics in Reykjavik. We also state and prove formally the correctness of these algorithms.

1 Introduction

Genetic linkage analysis is a well established method for studying the relationship between the pattern of the occurrence of a given disease and the inheritance pattern of certain genes in given families. In this sense genetic analysis of human traits is substantially different from the analysis of experimental plant and animal traits in that the observed pedigrees must be taken as given, but are not controlled as for the experimental organisms.

In what follows we give a short survey of the biological model the method is based on. The genetic material, i.e. the *genome*, of an organism is represented in the DNA (deoxyribonucleic acid) which is a chain of simpler molecules, *bases*. The genetic material is organized in the organism in such a way that each cell has a few very long DNA molecules, called *chromosomes*. Human beings are diploids, meaning that every individual carries two copies of each chromosome. According to Mendel's law, at each position or *locus*, there are equal probabilities of the genetic material of each of the two chromosomes of a parent being transmitted to a child. Before the parents pass their chromosomes on to their offspring, the chromosomes mix in a process called *meiosis*. The mixing involves large chunks of each chromosomes being transmitted intact, the transmission only being interrupted by transmissions of large chunks of the other chromosome. Where the transmission moves between the two chromosomes a *crossover* is said to have occurred. If an odd number of crossovers occurs between two loci, then

genetic material from different chromosomes is transmitted and a *recombination* is said to have occurred.

The genetic distance between two loci is defined as the expected number of crossovers that will occur in a single meiosis between the two loci. The unit of genetic distance is called a *Morgan*. The autosomal human chromosomes are between .7 and 2.7 Morgans in length. The *recombination fraction* between two loci is defined as the probability of a recombination occurring between them. It is commonly assumed that crossovers follow a Poisson process, in which case a simple mapping between genetic distance and recombination fractions exists, called the Haldane mapping function. Two loci are said to be *linked* if the recombination fraction between them is less than half. It is worth noting that loci on different chromosomes is unlinked. For more information about genetics we suggest the reader to consult [6].

The dependence between the transmissions statuses of loci that are physically, and therefore genetically, close in the genome is the basis of *linkage analysis*. The aim of linkage analysis is to identify loci where the *inheritance pattern* deviates substantially from Mendelian inheritance and this dependence allows the transmissions of the whole genome to be studied accurately with information only available from a small number of loci.

Linkage analysis is performed on a *pedigree*. The set of members V of each pedigree is divided into the set of founders (F), that do not have parents belonging to the pedigree, and non-founders (N), where both parents belong to it. (We can always assume that either both or neither of the parents of the individuals belong to the pedigree as we can always add the missing one.) When linkage analysis is performed, a number of known loci or *markers* are chosen. A marker is a locus where the set of possible states or *alleles*, and the frequency of each of these alleles, are known in the population. The recombination fraction between these markers must be known too. Such a set of markers with the information described above is referred to as a *marker map*.

Next those individuals of the pedigree that are accessible are *genotyped* for each of these markers. This results in a sequence of allele pairs, one for each marker under investigation. As one cannot directly observe from which parent each of the allele comes, i.e. the *phase* of the alleles is not observable, these pairs are unordered. Because the phase of the genotype is in general not known and because not all individuals in a pedigree can be genotyped (some of them might be unavailable for analysis), the precise inheritance pattern, i.e. which alleles are inherited from which parents, is represented by a probability distribution.

If only one marker is investigated at the time, independently of other loci, we talk about *single point analysis*. This amounts to calculating the probability of a given inheritance pattern at a given marker, given the genotypes at that particular marker. The genotypes of the markers do in general not give complete information about transmissions. Therefore, to get as precise probability distributions over the inheritance patterns as possible, the information about the neighbor genotype is also taken into account and we calculate the probability of a given inheritance pattern at a given marker given the genotypes for all markers. In this case we talk about *multi point analysis*. The multi point probabilities are derived from the single point ones by taking the probability of recombination into account.

Most contemporary methods for doing multi point linkage analysis are based either on an algorithm by Elston and Stewart [3] or the Lander-Green algorithm [12]. Based on the Elston-Stewart algorithm VITESSE is a state of the art software package and with the most recent advancements it can handle approximately 6-9 markers in a moderately large pedigree [14]. There are a number of software packages that use the Lander-Green algorithm and do non-parametric linkage analysis (NPL), notably GENEHUNTER [9], Allegro [5] and most recently MERLIN [1]. The most recent advances enable the analysis of pedigrees of approximately 20-30 bits [10,1].

In this paper we focus on the Lander-Green approach and we will only deal with the first phase of the multi point analysis by focusing on algorithms for calculating the probability distribution for inheritance patterns in single point analysis. These algorithms consist of two steps. First, for each inheritance pattern (i.e. the way the alleles are passed from the founders through the pedigree), the algorithms calculate the set of assignments of alleles values to the founder alleles that are compatible with it and the given genotype information. Then, based on the output of the first part, the algorithms calculate the probability distribution of the inheritance patterns.

In what follows we give a precise definition of most of the concepts emphasized above, that hopefully captures the essence of the standard informal ones. Then we describe, in terms of our framework, the two most commonly used algorithms for such founder allele assignment calculations in linkage analysis. The first one is a part of the GENHUNTER software package developed at the Whitehead Institute at MIT and is developed by Kruglyak et al [8]; the second one occurs in the software package ALLEGRO provided by Gudbjartsson et al [5] at DeCode Genetics in Reykjavik. We also state and prove formally the correctness of these algorithms. As far as we know, no similar formalization and correctness proofs for these algorithms have been given before. .

2 Pedigrees

We define a *pedigree* as a tree-like structure where the nodes represent the *members* of the pedigree and the arcs represent the parental relationships. As each individual only has one mother and one father, these relationships are represented as a function, the *parental* function. As explained in the introduction, each member of the pedigree is assumed to be either a *founder* of the pedigree where neither of his parents belongs to the pedigree, or a *non-founder* where both parents belong to it. In the definition to follow, if $F : A \times B \rightarrow C$ we write F_a for the function $y \mapsto F(a, y)$. Furthermore, following standard practice, for $f : A \rightarrow B$ and $A' \subseteq A$ we write $f(A')$ for $\{f(a) | a \in A'\}$.

Definition 1 (Pedigree). A pedigree consists of a 4-tuple $P = \langle V, F, \phi \rangle$ where the following holds:

- V is a finite set of members and $F \subseteq V$ is a set of founders of the pedigree.
- $\phi : (V \setminus F) \times \{p, m\} \rightarrow V$ is the parental function where
 - $\phi_p(V \setminus F) \cap \phi_m(V \setminus F) = \emptyset$ (nobody can be both a mother and a father) and
 - $\forall n \in V. (n, n) \notin (\phi_p \cup \phi_m)^+$, where $+$ is the transitive closure operator on relations (a member of the family is never its own ancestor).

The set $N = V \setminus F$ is referred to as the set of non-founders of the pedigree.

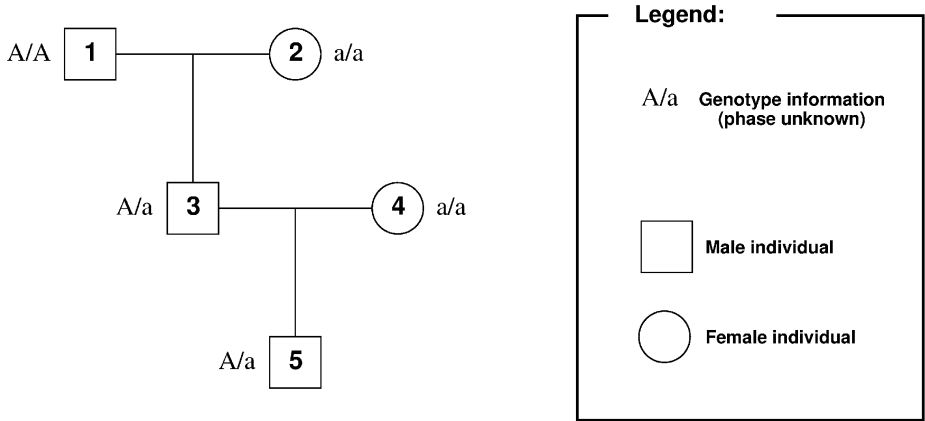


Fig. 1. An example pedigree with genotype information for which the phase is unknown

In what follows we assume a fixed pedigree $P = \langle V, F, \phi \rangle$ (unless stated otherwise).

Example 1. Below we give a formal representation of the example pedigree in Figure 1.

$$\begin{aligned}
 P &= \langle V, F, \phi \rangle \text{ where} \\
 V &= \{1, 2, 3, 4, 5\}, F = \{1, 2, 4\}, \\
 \phi_m(3) &= 2, \phi_p(3) = 1, \phi_m(5) = 4, \text{ and } \phi_p(5) = 3.
 \end{aligned}$$

Definition 2 (Genetic Material as Bits). We define the genetic material or the bits of a pedigree P as $\text{Bit} = V \times \{p, m\}$. We let (n, m) denote the maternal bit and (n, p) the paternal bit of the individual n . The bits of n , Bit_n are given by $\{(n, m), (n, p)\}$. We refer to the sets $\text{Bit}_N = N \times \{p, m\}$ and $\text{Bit}_F = F \times \{p, m\}$ as the non-founder bits and the founder bits of the pedigree, respectively.

An *inheritance pattern* indicates whether the individual inherits the paternal or maternal bit from each of his parents. If we fix the order of the set Bit , the inheritance pattern can be represented as an *inheritance vector*. In what follows, for a natural number K , we use the notation $[K] = \{1, 2, \dots, K\}$.

Definition 3 (Inheritance Pattern and Vector). An inheritance pattern i for some pedigree P is a function $i : \text{Bit}_N \rightarrow \{p, m\}$. An inheritance vector is a mapping $v : [2|N|] \rightarrow \{p, m\}$.

Obviously, for a fixed bit order, there is a 1-1 correspondence between inheritance patterns and vectors. Usually we let i, i_1, i' etc. denote an inheritance pattern and u, v, w etc. inheritance vectors.

Example 2. In the running Example 1 on page 375, an inheritance pattern could be (see Figure 2)

$$i(3, p) = p, i(3, m) = m, i(5, p) = p, \text{ and } i(5, m) = p.$$

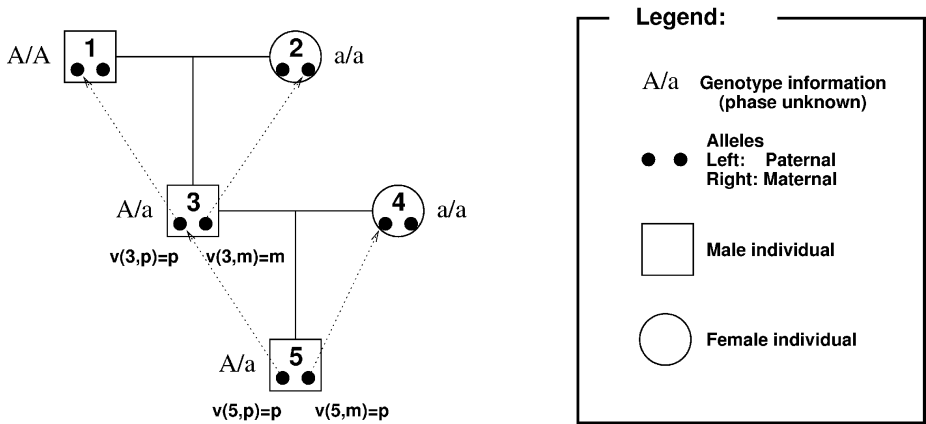


Fig. 2. A genotyped pedigree with a given inheritance pattern (indicated by the dotted lines)

From a given inheritance pattern, for each non founder bit, it is possible to trace from which founder bit it is inherited. This can be done by means of the function *Source* defined below.

Definition 4. For each inheritance pattern i we define a function $\text{Source}^i : \text{Bit} \rightarrow \text{Bit}_F$ by

$$\text{Source}^i(n, b) = \begin{cases} (n, b) & \text{if } n \in F \\ \text{Source}^i(\phi(n, b), i(n, b)) & \text{otherwise.} \end{cases}$$

We write $\text{Source}^i(n)$ as a shorthand for $\{\text{Source}^i(n, m), \text{Source}^i(n, p)\}$.

As mentioned earlier the genetic material comes in slightly different versions which we refer to as *alleles*. To capture this, from now on we assume a fixed set $\mathbf{A}$ of alleles. Furthermore we define *allele assignment* and *founder allele assignment* as assignments of alleles to the bits and the founder bits respectively.

Definition 5 (Allele Assignment). An allele assignment is a function that maps each bit in Bit to an allele: $\mathcal{A} : \text{Bit} \rightarrow \mathbf{A}$. A founder allele assignment is a function that maps each founder bit to an allele: $\mathcal{F} : \text{Bit}_F \rightarrow \mathbf{A}$.

Definition 6 (Genotype Assignment). The set of genotypes over $\mathbf{A}$, $\text{Geno}(\mathbf{A})$, is defined as the family of subsets of $\mathbf{A}$ that contain no more than two elements. Genotype assignment is a partial function $\mathcal{G} : V \rightarrow \text{Geno}(\mathbf{A})$, that associates a genotype to (some of the) members of the pedigree. The domain, $\text{dom}(\mathcal{G})$, of the function is referred to as the set of genotyped members of the pedigree. If $v \in \text{dom}(\mathcal{G})$ we say that n is genotyped (at that marker).

Remark 1. Note that in the definition above, a genotype assignment assigns an *unordered* pair of elements to members of the family. This indicates that the phase of the alleles is unknown. If a pedigree member is *homozygous* at a given locus, i.e. the same allele is assigned to both of its bits, the function $\mathcal{G}$ returns a singleton set.

Example 3. The pedigree in Figure1 shows genotype information for all individuals. Thus the set of genotyped individuals $\text{dom}(\mathcal{G})$ covers the complete set V of family members. The genotype of family member nr. 5 is $\mathcal{G}(5) = \{A, a\}$.

In the definition below, following standard practice, we assume that the definition of $\mathcal{F}$ extends to sets in a point wise manner, i.e. if $M \subseteq \text{Bit}$ then $\mathcal{F}(M) = \{\mathcal{F}(g) | g \in M\}$.

Definition 7 (Compatibility). A founder allele assignment $\mathcal{F}$ is said to be compatible with the genotype assignment $\mathcal{G}$ and the inheritance pattern i if the following holds:

$$\forall n \in \text{dom}(\mathcal{G}) . \mathcal{F}(\text{Source}^i(n)) = \mathcal{G}(n) .$$

We let $\text{Comp}_P(i, \mathcal{G})$ denote the set of such founder allele assignments. If $\text{Comp}_P(i, \mathcal{G}) \neq \emptyset$ then i is said to be compatible with $\mathcal{G}$.

3 Algorithms for the Single Point Probability Calculations

A genotype (at a fixed marker) can be represented a set of pairs of the form $(n, \{a_1, a_2\})$, where n is a member of the family and a_1 and a_2 are alleles. Intuitively such a pair is supposed to indicate that these two alleles have to be assigned to the bits of n , (n, m) and (n, p) ; we just do not know which is assigned to which as phase can not be decided directly. If the inheritance pattern i is known, we may find out from which founder bits the bits of n are originated. Therefore, in this case we may consider the genotype assignment as an implicit specification of a founder allele assignment as the bits of n must be assigned the same values as the founder bits they came from.

As explained in the introduction we will describe in detail two algorithms that occur to be among the most commonly applied ones in connection with linkage analysis. The first one is the algorithm employed in the linkage analysis tool GENHUNTER and was developed by Kruglyak et al [8]; the second one is the algorithm of the software tool Allegro at DeCode Genetics, Reykjavik, and is provided by Gudbjartsson et al [5].

Before we present the algorithms we will explain the notation they rely on. In the heads of the algorithms we have the set I_P , the set of all inheritance patterns over P . Both algorithms take as input a pedigree P and genotype assignment $\mathcal{G}$ and return a mapping that, to each inheritance pattern $i \in I_P$, assigns a pair $(\mathcal{S}, \mathcal{P})$, that can be considered as an implicit representation of the set of founder allele assignments which are compatible with P and $\mathcal{G}$ given i ; if the input is incompatible the algorithms return the element $\perp$. We refer to elements of the type described above as implicit founder allele assignment (ifa) and the set of such elements is denoted by IFA_P .

The components $\mathcal{S}$ and $\mathcal{P}$ represent the fully specified part of the assignment and the partially specified one respectively. The set $\mathcal{S}$ consists of elements of the form (f, a) , where f is a founder allele and a is an allele, which represents the set of all founder allele assignments that map f into a . The second set $\mathcal{P}$ consists of elements of the form $(\{f_1, f_2\}, \{a_1, a_2\})$ (for short we often write $(f_1 f_2, a_1 a_2)$ instead), which represents the set of assignments that either map f_1 to a_1 and f_2 to a_2 , or f_1 to a_2 and f_2 to a_1 . We let $\text{dom}(\mathcal{S})$ and $\text{dom}(\mathcal{P})$ denote the set of bits that occur in the sets $\mathcal{S}$ and $\mathcal{P}$ respectively. The interpretation described above can be expressed formally by the function Interpret as follows.

Definition 8. *The function Interpret is defined as follows:*

$$\begin{aligned} \text{Interpret}(f, a) &= \{\mathcal{A} \mid \mathcal{A}(f) = a\} \\ \text{Interpret}(f_1 f_2, a_1 a_2) &= \{\mathcal{A} \mid \mathcal{A}(f_1) = a_1 \text{ and } \mathcal{A}(f_2) = a_2, \\ &\quad \text{or } \mathcal{A}(f_1) = a_2 \text{ and } \mathcal{A}(f_2) = a_1\}, \\ \text{Interpret}(\mathcal{X}) &= \bigcap_{\alpha \in \mathcal{X}} \text{Interpret}(\alpha) \text{ for } \mathcal{X} = \mathcal{S}, \mathcal{P} \\ \text{Interpret}(\mathcal{S}, \mathcal{P}) &= \text{Interpret}(\mathcal{S}) \cap \text{Interpret}(\mathcal{P}), \\ \text{Interpret}(\perp) &= \emptyset, \end{aligned}$$

$\gamma_1, \gamma_2 \in \text{IFA}_P$ are said to be equivalent, $\gamma_1 \sim \gamma_2$, iff $\text{Interpret}(\gamma_1) = \text{Interpret}(\gamma_2)$.

Lemma 1. *If $(\mathcal{S}, \mathcal{P}) \sim (\mathcal{S}', \mathcal{P}')$ then $\text{dom}(\mathcal{S}) \cup \text{dom}(\mathcal{P}) = \text{dom}(\mathcal{S}') \cup \text{dom}(\mathcal{P}')$.*

Definition 9. *The ifa $(\mathcal{S}, \mathcal{P})$ is said to be fully reduced if it is either $\perp$ or following is satisfied:*

- $\text{dom}(\mathcal{S}) \cap \text{dom}(\mathcal{P}) = \emptyset$ and
- if $(\mathcal{S}, \mathcal{P}) \sim (\mathcal{S}', \mathcal{P}')$ then $\text{dom}(\mathcal{S}') \subseteq \text{dom}(\mathcal{S})$.

Lemma 2. – *For each element $\gamma \in \text{IFA}_P$ there exists exactly one fully reduced $\eta \in \text{IFA}_P$, which is equivalent to it. We refer to η as the normal form for γ .*
 – $\gamma \in \text{IFA}_P$ is incompatible iff its normal form is $\perp$.

In the light of the lemma above, an ifa in normal form that is different from $\perp$, can be written as $\{(f_1 \dots f_n, h_1 \dots h_l, ab), \dots\}$, where $a \neq b$ and the f 's and the h 's are all different. The intuitive meaning of this is that either all the f 's should be assigned to a and the h 's to b or the other way around.

Kruglyak's algorithm: The algorithm KRUGLYAK is straight forward and simply, given an inheritance pattern, builds the implicit founder allele assignments directly from the genotype information. It is divided into three phases. The first phase, BUILDFOUNDERS, builds the ifa from the genotype information for the founders. This phase is independent of the inheritance pattern and therefore it only needs to be gone through once. The second phase, BUILDFORNONFOUNDERS, extends, for each inheritance pattern, the ifa that was returned from the first phase, by adding to it the genotype information for the non founders. The final phase, REDUCE, returns the unique fully reduced ifa equivalent to the output of the previous phase. We will not describe the function REDUCE but only assume that it exists and satisfies the properties described above. Other parts of the algorithm can be seen below.

The algorithm KRUGLYAK builds a function from I_P to IFA_P for each element of I_P independently. As, by assumption, the function REDUCE preserves equivalence of ifa's, to show the correctness of the algorithm it is sufficient to show that

$$\begin{aligned} \text{Interpret}(\text{BUILDFORNONFOUNDERS}(i, \text{BUILDFORFOUNDERS}(\mathcal{G}), \mathcal{G})) \\ = \text{Comp}_P(i, \mathcal{G}). \end{aligned}$$

Algorithm 1 KRUGLYAK(in: $\mathcal{G}, P$; out: $(\mathcal{A}, \mathcal{P}) \in I_P \longrightarrow IFA_P$)

```

 $\mathcal{P} \leftarrow \text{BUILDFORFOUNDERS}(\mathcal{G})$ 
for  $i \in I_P$  do
   $\mathcal{P}' \leftarrow \text{BUILDFORNONFOUNDERS}(i, \mathcal{P}, \mathcal{G})$ 
   $(\mathcal{A}, \mathcal{P}) \leftarrow \text{REDUCE}(\mathcal{P}')$ 
end for

```

Algorithm 2 BUILDFORFOUNDERS(in: $\mathcal{G}$; out: $\mathcal{P}$)

```

 $\mathcal{P} \leftarrow \emptyset$ 
for  $n \in \text{dom}(\mathcal{G}) \cap F$  do
   $\mathcal{P} \leftarrow \mathcal{P} \cup \{(A(n), \mathcal{G}(n))\}$ 
end for

```

Algorithm 3 BUILDFORNONFOUNDERS(in: $i, \mathcal{P}, \mathcal{G}$; out: $\mathcal{P}'$)

```

 $\mathcal{P}' \leftarrow \mathcal{P}$ 
for  $n \in \text{dom}(\mathcal{G}) \cap N$  do
   $\mathcal{P}' \leftarrow \mathcal{P}' \cup \{(F^i(n), \mathcal{G}(n))\}$ 
end for

```

This in turn follows easily by inspecting the functions involved.

Unlike the algorithm KRUGLYAK, the algorithm by Gudbjartsson et al discovers incompatibilities as soon as they occur. The theoretical considerations behind this approach can be described in the following lemma, where a subpedigree of a pedigree has the expected meaning.

Lemma 3. *If $(P, \mathcal{G}, i)$ is compatible, $P' = (V', F', \phi')$ is a subpedigree of P and i' and $\mathcal{G}'$ are i and $\mathcal{G}$ restricted to $N' \cup F'$ respectively, then $(P', \mathcal{G}', i')$ is also compatible.*

Intuitively Lemma 3 says that any substructure of a compatible structure of the form $(P, \mathcal{G}, i)$ must be compatible too. Turning this reasoning the other way around we get that if such a structure is incompatible, then all extensions of it remain incompatible.

Gudbjartsson's Algorithm: The algorithm GUDBJARTSSON works by first calling the algorithm BUILDFORFOUNDERS that investigates the initial substructure only consisting of the founder alleles and the corresponding genotype assignment. Then it calls the recursive algorithm TRAVERSE TREE that gradually extends this structure by, at each recursive call, adding a new member to the pedigree. At the same time it extends both the genotype assignment with this new member and the inheritance pattern in both possible ways. For this to work we preassume an order o of the non founders of the input pedigree that respects the family hierarchy. More precisely we assume that if $o(n_1) < o(n_2)$ then $(n_2, n_1) \in (\phi_p \cup \phi_m)^+$. Furthermore an inheritance vector is represented by a string $v \in \{m, p\}^{2K}$, where K is the number of non founders of the family. If $v = b_1^p b_1^m \cdots b_K^p b_K^m$ this can be interpreted by $v(n_j, p) = b_j^p$ and $v(n_j, m) = b_j^m$. The algorithm of Gudbjartsson et al is given in Algorithms 4–6.

To prove the correctness of the algorithm GUDBJARTSSON we show that it satisfies the invariance conditions states in the following lemma.

Algorithm 4 GUDBJARTSSON(**in**: $\mathcal{G}, P$; **out**: $\Delta \in I_P \longrightarrow IFA_P$)

 $(\mathcal{S}, \mathcal{P}) \leftarrow \text{BUILDFORFOUNDERS}(\mathcal{G}, P)$
 $\text{TRAVERSE TREE}(\varepsilon, (\mathcal{S}, \mathcal{P}), \mathcal{G}, P)$

Algorithm 5 TRAVERSE TREE(**in** : $v, (\mathcal{S}, \mathcal{P}), \mathcal{G}, P$; **out** : Δ)

if $|v| = |N|$ **then**
 return $(v, (\mathcal{S}, \mathcal{P}))$
else
 for $(b_p, b_m) \in \{m, p\}^2$ **do**
 $v' \leftarrow vb_p b_m$
 $\gamma \leftarrow (\mathcal{S}, \mathcal{P})$
 $j \leftarrow |v'|$
 if $n_j \in \text{dom}(\mathcal{G})$ **then**
 $\gamma \leftarrow \text{INSERT}(\text{Source}^{v'}(n), \mathcal{G}(n), \gamma)$
 if $\gamma = \perp$ **then**
 return $(v', \perp)$
 else
 $\text{TRAVERSE TREE}(v', \gamma, \mathcal{G}, P)$
 end if
 else
 $\text{TRAVERSE TREE}(v', \gamma, \mathcal{G}, P)$
 end if
 end for
end if

Lemma 4. Let P_l and $\mathcal{G}_l$ denote the subpedigree and genotype assignment obtained by only considering the set, $\{n_1, \dots, n_l\}$, of pedigree members. Let furthermore Interpret_{P_l} be the interpretation function w.r.t. P_l . Then we have:

- If γ is fully reduced, then $\text{Insert}(ff', aa', \gamma)$ is also fully reduced.
- If $\text{Interpret}_{P_l}(\gamma) = \text{Comp}_{P_l}(v, \mathcal{G}_l)$, then
 - $\text{Interpret}_{P_{l+1}}(\gamma) = \text{Comp}_{P_{l+1}}(vb_p b_m, \mathcal{G}_{l+1})$ for $b_p, b_m \in \{p, m\}$, if $n \notin \text{dom}(\mathcal{G})$.
 - $\text{Interpret}_{P_{l+1}}(\text{Insert}(\mathcal{F}^{vb_p b_m}(n), \mathcal{G}(n), \gamma)) = \text{Comp}_{P_{l+1}}(vb_p b_m, \mathcal{G}_{l+1})$ for $b_p, b_m \in \{p, m\}$, if $n \in \text{dom}(\mathcal{G})$.

The following corollary states exactly the correctness of the algorithm and follows directly from Lemma 4.

Corollary 1. – If the algorithm GUDBJARTSSON outputs (v, γ) , where $\gamma \neq \perp$, then $|v| = |N|$ and $\text{Interpret}(\gamma) = \text{Comp}_P(v, \mathcal{G})$.
 – If it outputs $(v, \perp)$, then $(P_{|v|}, \mathcal{G}_{|v|}, v)$ and all its extensions are incompatible, (i.e. the inheritance vector obtained as it is incompatible with P and $\mathcal{G}$ for all $t \in \{m, p\}^{2(|V|-|v|)}$).

Probability Calculations for the Single Point Case. In this section we show how the single point probabilities, i. e. the probability distribution over inheritance patterns at a

Algorithm 6 INSERT(in : $(\{f, f'\}, \{a, a'\}), (\mathcal{S}, \mathcal{P}), \mathcal{G}, P$; out : γ)

```

 $\gamma \leftarrow (\mathcal{S}, \mathcal{P})$ 
if  $f = f'$  and  $a = a'$  then
  if  $(f, a) \in \mathcal{S}$  then
    do nothing
  else if  $(f\phi, a\alpha) \in \mathcal{P}$  for some  $\phi, \alpha$  then
    remove  $(f\phi, a\alpha)$  from  $\mathcal{P}$ 
    add  $(f, a)$  and  $(\phi, \alpha)$  to  $\mathcal{S}$ 
  else if  $f \notin \mathcal{S} \cup \mathcal{P}$  then
    add  $(f, a)$  to  $\mathcal{S}$ 
  else
     $\gamma \leftarrow \perp$ 
  end if
else if  $f \neq f'$  then
  if  $(f, a), (f', a') \in \mathcal{S}$  then
    do nothing
  else if  $(f, a) \in \mathcal{S}, (f'\phi, a'\alpha) \in \mathcal{P}$  for some  $\phi, \alpha$  then
    remove  $(f'\phi, a'\alpha)$  from  $\mathcal{P}$ 
    add  $(f', a'), (\phi, \alpha)$  to  $\mathcal{S}'$ 
  else if  $(f\phi, a\alpha), (f'\phi', a'\alpha') \in \mathcal{P}$  then
    remove  $(f\phi, a\alpha), (f'\phi', a'\alpha')$  from  $\mathcal{P}$ 
    add  $(f, a), (f', a'), (\phi, \alpha), (\phi', \alpha')$  to  $\mathcal{S}$ 
  else if  $(ff', aa') \in \mathcal{P}'$  then
    do nothing
  else if  $(f, a) \in \mathcal{S}, f' \notin \mathcal{S} \cup \mathcal{P}$  then
    add  $(f', a')$  to  $\mathcal{S}$ 
  else if  $(f\phi, a\alpha) \in \mathcal{P}, f' \notin \mathcal{S} \cup \mathcal{P}$  then
    remove  $(f\phi, a\alpha)$  from  $\mathcal{P}$ 
    add  $(f, a), (\phi, \alpha), (f', a')$  to  $\mathcal{S}$ 
  else if  $f, f' \notin \mathcal{S} \cup \mathcal{P}$  and  $a \neq a'$  then
    add  $(ff', aa')$  to  $\mathcal{P}$ 
  else if  $f, f' \notin \mathcal{S} \cup \mathcal{P}$  and  $a = a'$  then
    add  $(f, a), (f', a')$  to  $\mathcal{S}$ 
  else
     $\gamma \leftarrow \perp$ 
  end if
else
   $\gamma \leftarrow \perp$ 
end if

```

given marker, given a genotype assignment at that marker, is obtained from the sets of compatible founder allele assignments associated to each inheritance pattern.

To calculate the single point probabilities $p(v|\mathcal{G})$ for all inheritance patterns v , we first note that by Bayes' theorem

$$p(v|\mathcal{G}) = \frac{p(\mathcal{G}|v)}{p(v)}p(\mathcal{G}) = Kp(\mathcal{G}|v),$$

where $K = \frac{p(\mathcal{G})}{p(v)}$ is independent of v . Therefore it is sufficient to calculate $p(\mathcal{G}|v)$ for all v and replace the output $(s, p(\gamma))$ in the algorithms above by the results. As explained earlier, for a given v , the pedigree P will have the genotype given by $\mathcal{G}$ exactly for founder allele assignments from the set $Comp_P(v, \mathcal{G})$. This set, in turn, is uniquely decided by the pair $(\mathcal{S}, \mathcal{P})$ that is associated to v by the algorithms above when they are run on $\mathcal{G}$ as input. In the calculations of $p(\mathcal{S}, \mathcal{P})$ we can assume that the allele frequency is independent of the founders. Thus, if we let π denote the frequency distribution over A in the given population, this probability can be obtained as follows:

- $p(g, a) = \pi(a)$, $p(f_1 f_2, a_1 a_2) = 2\pi(a_1)\pi(a_2)$,
- $p(\mathcal{X}) = \prod_{\alpha \in \mathcal{X}} p(\alpha)$ for $\mathcal{X} = \mathcal{S}, \mathcal{P}$,
- $p(\mathcal{S}, \mathcal{P}) = p(\mathcal{S})p(\mathcal{P})$, $p(\perp) = 0$.

4 Conclusion

In this paper we have defined a mathematical model that describes genetic inheritance in pedigrees. We have used this model to describe two algorithms which are widely used for calculating the complete set of compatible founder allele assignments in connection with linkage analysis and to reason about their correctness.

It is immediately clear, but also shown by means of software testing, that Gudbjartsson's algorithm performs much better than that of Kruglyak. However, it has been shown [2] that the problem of checking whether a pair consisting of a pedigree and some genotype information is at all compatible (i.e. if *any* inheritance pattern is compatible with it) is NP-complete. This means that it is not very likely that there exists any polynomial time algorithm that investigates the compatibility and that the problem considered here is most likely inherently complicated.

The main contribution of the paper is to provide a formal framework to express and reason about the process of linkage analysis. In particular we state and proof the correctness of known algorithms in the area. As far as we know these algorithms have not been analysed and proven correct before.

The results reported here have already served as the theoretical background for further investigations that focus on the full multi point calculations. The results of these studies are reported in [4].

References

1. Abecasis, G. R., Cherny, S. S., Cookson, W. O., Cardon, L. R. Merlin – rapid analysis of dense genetic maps using sparse gene flow trees. *Nature Genetics* 30:97–101 (2001)
2. Aceto, L., Hansen, J. A., Ingólfssdóttir, A., Johnsen, J., Knudsen, J. Checking consistency of pedigree information is NP-complete (Preliminary report) BRICS Report Series RS-02-42, October 2002. Available at <http://www.brics.dk/RS/02/42/>
3. Elston, R. C., Stewart, J. A general model for the genetic analysis of pedigree data. *Hum Hered* 21:523–542 (1971)
4. Gudbjartsson, D. F., Gunnarsson, G., Ingólfssdóttir, A., Stefansson, S., Thorvaldsson, Th. BDD-based algorithms in genetic linkage analysis. BRICS Report Series RS-03-??, Available at <http://www.brics.dk/RS/03/>.

5. Gudbjartsson, D. F., Jonasson, K., Frigge, M., Kong, A. Allegro, a new computer program for multipoint linkage analysis. *Nature Genetics* 25:12–13 (2000)
6. Lynn B. Jorde, John C. Carey, Michael J. Bamshad, and Raymond L. White. *Medical Genetics*. Mosby, 1999.
7. Klug, W. S., Cummings, M. R. *Concepts of genetics 5th edition*. Prentice Hall, 1997.
8. Kruglyak, L., Daly, M. J., Reeve-Daly, M. P., Lander, E. S. Parametric and nonparametric linkage analysis: A unified multipoint approach. *Am. J. Hum. Genet.*, 58:1347–1363, 1996.
9. Kruglyak, L., Lander, E. S. Faster multipoint linkage analysis using fourier transforms. *Journal of Computational Biology*, 5(1):7, 1998.
10. Markianos, K., Daly, M. J., Kruglyak, L. Efficient multipoint linkage analysis through reduction of inheritance space. *American journal of human genetics*, 68:963–977, 2001.
11. Lange, K. *Mathematical and statistical methods for genetic analysis*. Springer, 1997.
12. Lander, E. S., Green, P. Construction of multilocus genetic linkage maps in humans. *Proc. Natl. Acad. Sci.*, 84:2363–2367, 1987.
13. Ott, J. *Analysis of human genetic linkage, third edition*. The Johns Hopkins University Press, 1999.
14. O’Connell, JR. Rapid multipoint linkage analysis via inheritance vectors in the Elston-Stewart algorithm. *Hum Hered* 51:226–240, 2001)

A Self-adaptive Model for Selective Pressure Handling within the Theory of Genetic Algorithms

Michael Affenzeller and Stefan Wagner

Institute of Systems Science
Systems Theory and Information Technology
Johannes Kepler University
Altenbergerstrasse 69
A-4040 Linz - Austria
`{ma,sw}@cast.uni-linz.ac.at`

Abstract. In this paper we introduce a new generic selection method for Genetic Algorithms. The main difference of this selection principle in contrast to conventional selection models is given by the fact that it considers not only the fitness of an individual compared to the fitness of the total population in order to determine the possibility of being selected. Additionally, in a second selection step, the fitness of an offspring is compared to the fitness of its own parents. By this means the evolutionary process is continued mainly with offspring that have been created by advantageous combination of their parents' attributes. A self-adaptive feature of this approach is realized in that way that it depends on the actual stadium of the evolutionary process how many individuals have to be created in order to produce a sufficient amount of 'successful' offspring. The experimental part of the paper documents the ability of this new selection operator to drastically improve the solution quality. Especially the bad properties of rather disadvantageous crossover operators can be compensated almost completely.

1 Introduction

Genetic Algorithms (GAs) are search and optimization algorithms which are based on the fundamentals of natural evolution. In Fig. 1; we represent Evolutionary Algorithms in relation to other search techniques with special attention directed to Genetic Algorithms.

The basic principles of GAs were first presented by Holland [9]. Since that time GAs have been successfully applied to a wide range of problems including multimodal function optimization, machine learning, and the evolution of complex structures such as neural networks. An overview of GAs and their implementation in various fields is given by Goldberg [7] or Michalewicz [11].

When applying GAs to large and complex problems, one of the most frequent difficulties is premature convergence. Roughly speaking, premature convergence occurs when the population of a GA reaches such a suboptimal state that the

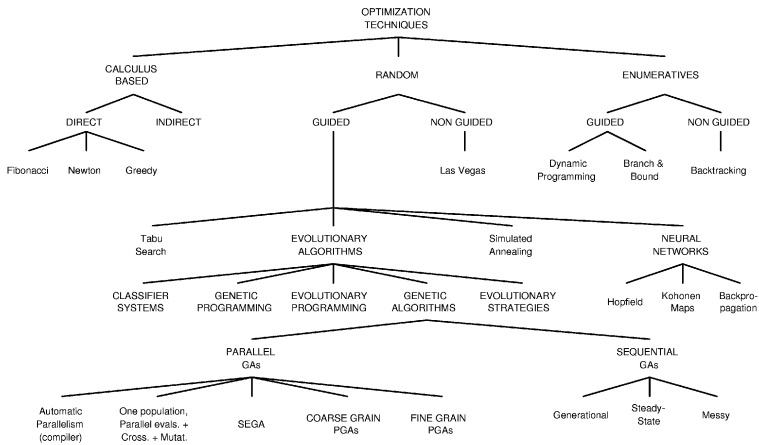


Fig. 1. Taxonomy of optimization techniques.

genetic operators (crossover, mutation) can no longer produce offspring that outperform their parents. Various methods have been proposed to retard the unwanted effects of premature convergence. Among others, these include modifications in the recombination procedure, in the selection procedure or in the fitness assignment (e.g. [15]). However, the effects of all these methods vary with different problems and their implementations.

A critical problem when studying premature convergence is the identification of its occurrence and its extent. In some contributions the difference between the average and maximum fitness is used as a value to measure premature convergence and the crossover and mutation probabilities are varied adaptively according to this measurement. Also the term population diversity has been used in many papers (e.g. [14]) to study premature convergence where the decrease of population diversity is considered as the primary reason for premature convergence. Therefore, a very homogeneous population, i.e. little population diversity, is the major reason for a Genetic Algorithm to prematurely converge. However, so far there exists little effort in performing a generic analysis of population diversity. But it is well known that the degree of population diversity is controllable by selective pressure.

In case of standard GAs selective pressure can only be influenced by the population size, by the choice of the selection mechanism and the operators as well as by the corresponding parameters. As these controls are quite complicated and also influence other characteristics of the GA we have modified the basic concept of GAs in a way that allows a simple and direct steering of selective pressure with only one additional parameter ([2], [4]). In a way quite similar to the (μ, λ) Evolution Strategy (ES) a virtual population of a size not smaller than the actual population size is introduced. Like in a standard GA the individuals of this virtual population are generated by selection, crossover and mutation. The actual new generation, i.e. the population that provides the heritable information

for the remaining search process, is then built up with the best members of the virtual population. The greater the virtual population size is adjusted in comparison to the actual population size, the higher is the actual setting of selective pressure. In case of an equal sized virtual and actual population the advanced algorithm operates completely analogical with practically the same running time as the underlying GA. With this enhanced model it is already possible to achieve results clearly superior to the results of a comparable GA. Furthermore, this model allows a natural and intuitive formulation of further new biologically inspired parallel hybrid GA approaches ([1], [3]).

Even if these new GA variants are able to significantly outperform comparable GAs and similar heuristic optimization techniques in terms of global solution quality, a major drawback is the time-consuming job of parameter tuning. As GAs implement the idea of evolution, and as evolution itself must have evolved to reach its current state of sophistication, it is only natural to expect adaptation not only to be used for finding solutions to a problem, but also for tuning the algorithm to the particular problem. Therefore, this paper discusses new generic aspects of self-adaptation - especially for selective pressure steering.

In doing so we adopt the $\frac{1}{5}$ success rule postulated by Rechenberg for Evolution Strategies into our enhanced selection model for GAs. With this strategy it becomes possible to steer the evolutionary search process in such a way that sufficiently enough population diversity is maintained for the GA not to prematurely converge without the need to set up additional problem and implementation specific schedules for selective pressure steering as required in the implementations discussed in [2] and [4]. Furthermore, the algorithm is now able to automatically detect the phase when premature convergence effectively occurs which gives a reasonable termination criterion for our enhanced GA and, even more important, this self adaptive strategy for selective pressure steering can be used as a detector for an appropriate reunification date of subpopulations for the massively parallel SEGA-algorithm ([1], [3]). Thus, these newly introduced aspects of self adaptation make the algorithm more user-friendly and save a lot of testing work and parameter adjustment when applying the certain GA-derivatives to various kinds of problems. The experimental part of the paper compares the obtained results to the results of a corresponding standard GA.

2 The Self-adaptive Selection Model

The basic idea of Evolutionary Algorithms is to merge the genetic information in a way that those building blocks will 'survive' during the evolutionary process which are essential for a global solution w.r.t. a given fitness function. In this context, the aim of selection is to choose those candidates for reproduction which are rather expected to contain the essential building-blocks. All popular selection mechanisms like roulette wheel, linear-ranking, or tournament selection fulfill this essential requirement, where the main difference of these strategies is given by their diverse selection pressure as described in [6]. However, it is a common property of all mentioned selection schemes that they consider only the fitness

value of the parents which are chosen for reproduction. Therefore, parents with above-average fitness values are selected for crossover with higher probability - but the quality of the children who are generated from the selected parents is not taken into account. Especially in case of artificial evolution it happens quite frequently that essential building blocks of the genetic information in the parent generation get lost in the reproduction process and are therefore no more available for the ongoing evolutionary process.

Inspired by Rechenberg's $\frac{1}{5}$ success rule for Evolution Strategies (e.g. described in [13]), we have developed an advanced selection model for Genetic Algorithms that allows self-adaptive control of selection pressure in a quite intuitive way:

In a first (conventional) selection step, parents are chosen for reproduction by roulette-wheel, liner-rank or some kind of tournament-selection like in the case of a normal Evolutionary Algorithm. The difference to conventional selection is that the offspring generated from the selected parents do not automatically become members of the next generation. In our new model the quality of reproduction is measured in a second step of selection by comparing the fitness of the child with the fitness values of its parents. Only if the fitness value of the generated child is 'better' than the fitness of its parents, the child is accepted in the next generation, i.e. members of the mating pool for the further evolutionary process.

However, the meaning of 'better' has to be explained in more detail: is a child better than its parents, if it surpasses the fitness of the weaker, the better, or is it in fact some kind of mean value of both? For this problem we have decided to introduce a cooling strategy similar to Simulated Annealing. Following the basic principle of Simulated Annealing we claim that an offspring only has to surpass the fitness value of the worse parent in order to be considered as 'successful' at the beginning and while evolution proceeds the child has to be better than a fitness value continuously increasing between the fitness of the weaker and the better parent. Like in the case of Simulated Annealing, this strategy effects a broader search at the beginning whereas at the end of the search process this operator acts in a more and more directed way.

The number of offspring that have to be created in that way depends on a predefined ratio-parameter ($SuccRatio \in [0, 1]$) giving the quotient of next generation members that have to outperform their own(!) parents. As long as this ratio is not fulfilled further children are produced. When the postulated ratio is reached, the rest of the next generation members are randomly chosen from the children that did not fulfill the fitness criterion. Within our new selection model we define selective pressure ($ActSelPress$) as the ratio of generated candidates to the population size. A default upper limit for selection pressure ($MaxSelPress$) gives a quite intuitive termination criterion: if it is no more possible to find a sufficient number of offspring that outperform their parents, premature convergence has occurred. Fig. 2 shows the operating sequence of the above described concepts.

By means of this novel selection strategy the appearance of clones is retarded and the bad properties of crossover operators are compensated - especially in

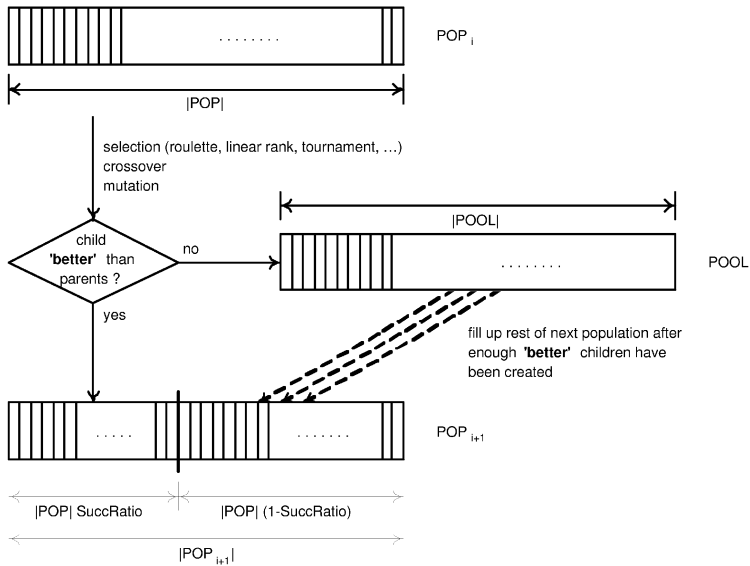


Fig. 2. Flowchart for embedding the new selection principle into a Genetic Algorithm.

case of rather critical crossover mechanisms as often applied in practical applications. As crossover results which do not mix their parents' genetic information advantageously are simply not considered, the ongoing evolutionary process is indeed proceeded with building blocks mainly derived from the parent generation. Thus, evolutionary search can be directed more efficiently without supporting the unwanted effects of premature convergence caused by too uniform solution candidates.

An additionally introduced concept suggests the appliance of multiple crossover operators simultaneously in order to roughly imitate the parallel evolution of a variety of species. This strategy seems very suitable for problems which consider more than one crossover operator - especially if the properties of the available operators may change as evolution proceeds. Furthermore, it is observable that the maintenance of population diversity is supported if more crossover operators are involved.

As an important property of the newly introduced methods it has to be pointed out that the corresponding GA is unrestrictedly included in this new variant of an Evolutionary Algorithm under special parameter settings. The experimental part analyzes the new algorithm for the Traveling Salesman Problem (TSP) which represents a very well documented instance of a multimodal combinatorial optimization problem. In contrast to all other evolutionary heuristics known to the authors that do not use any additional problem-specific information, we obtain solutions very close to the best-known solution for all considered benchmarks.

Algorithm 1 Standard Genetic Algorithm (SGA)

```

Initialize total number of iterations  $nrOfIterations \in \mathbb{N}$ 
Initialize size of population  $|POP|$ 
Produce an initial population  $POP_0$  of size  $|POP|$ 

for  $i = 1$  to  $nrOfIterations$  do
  Initialize next population  $POP_{i+1}$ 

  while  $|POP_{i+1}| \leq |POP|$  do
    Select two parents  $par_1$  and  $par_2$  from  $POP_i$ 
    Generate a new child  $c$  from  $par_1$  and  $par_2$  by crossover
    Mutate  $c$  with a certain probability
    Insert  $c$  into  $POP_{i+1}$ 
  end while
end for

```

3 An Algorithmic Description

Alg. 1 and Alg. 2 opposite the basic concept of a standard GA with the new GA that is equipped with the described self-adaptive selection mechanism in an algorithmic description. Even if the nomenclature 'Standard-GA' is not exactly standardized in GA literature we use this term for this version of a GA with which we qualitatively compare our new concepts.

4 Empirical Studies

Empirical studies with different problem classes and instances are widely considered as the most effective way to analyze the potential of heuristic optimization techniques like Evolutionary Algorithms. Even if a convergence proof similar to that of Simulated Annealing [8] may be possible, we are unfortunately confronted with the drawback that the number of states of the Markov Chain blows up from $|\mathcal{S}|$ to $|\mathcal{S}^{POP}|$, thus limiting the computational tractability to small problems and very small population sizes.

In our experiments, all computations are performed on a Pentium 4 PC with 1 GB of RAM. The programs are written in the programming language C#. For the tests we have selected the Traveling Salesman Problem (TSP) as a well documented instance of a typical multimodal combinatorial optimization problem. We have tested the new concepts on a selection of symmetric as well as asymmetric TSP benchmark problem instances taken from the TSPLIB [12] using updated results¹ for the best or at least best-known solutions. In all experiments, the results are represented as the relative difference to the best-known solution defined as $relativeDifference = (\frac{Result}{Optimal} - 1) \cdot 100\%$.

¹ Updates for the best-(known) solutions can be found for example on <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>

Algorithm 2 Genetic Algorithm with New Selection

```

Initialize total number of iterations  $nrOfIterations \in \mathbb{N}$ 
Initialize actual number of iterations  $i = 0$ 
Initialize size of population  $|POP|$ 
Initialize success ratio  $SuccRatio \in [0, 1]$ 
Initialize maximum selection pressure  $MaxSelPress \in [1, \infty[$ 
Initialize actual selection pressure  $ActSelPress = 1$ 
Initialize comparison factor  $CompFact = 0$ 
Produce an initial population  $POP_0$  of size  $|POP|$ 

while  $(i < nrOfIterations) \wedge (ActSelPress < MaxSelPress)$  do
  Initialize next population  $POP_{i+1}$ 
  Initialize pool for bad children  $POOL$ 

  while  $(|POP_{i+1}| < |POP| \cdot SuccRatio) \wedge (|POP_{i+1}| + |POOL| < |POP| \cdot$ 
     $MaxSelPress)$  do
    Generate a child from the members of  $POP_i$  due to their fitnesses by crossover
    and mutation

    Compare the fitness of the child  $c$  to the fitnesses of its parents  $par_1$  and  $par_2$ 
    (w.l.o.g. assume that  $par_1$  is fitter than  $par_2$ )
    if  $f_c \leq f_{par_2} + |f_{par_1} - f_{par_2}| \cdot CompFact$  then
      Insert child into  $POOL$ 
    else
      Insert child into  $POP_{i+1}$ 
    end if
  end while
   $ActSelPress = \frac{|POP_{i+1}| + |POOL|}{|POP|}$ 

  Fill up the rest of  $POP_{i+1}$  with members from  $POOL$ 
  while  $|POP_{i+1}| \leq |POP|$  do
    Insert a randomly chosen child from  $POOL$  into  $POP_{i+1}$ 
  end while
  Increase  $CompFactor$  according to the used annealing strategy
   $i = i + 1$ 
end while

```

The first feature that is examined in the experimental part is the comparison of a GA with the new selection mechanism against the corresponding GA with conventional selection for a variety of crossover operators for the TSP [11], [10]. Tab. 2 shows these comparisons with roulette-wheel as the first selection step of our new model (respectively as the general selection mechanism in case of the standard-GA).

The evaluation of the results of Tab. 2 shows the remarkable effect that also crossover operators that are considered as rather unsuitable for the TSP [10] achieve quite good results in combination with the new selection model. The reason for this is given by the fact that in our selection-principle only children that have emerged from a good combination of their parents' attributes are

Table 1. Parameter set used in the model of Table 2.

<i>generations</i>	5000
<i>population size</i>	200
<i>mutation rate</i>	0.05
<i>success ratio</i>	0.6

Table 2. Comparison of SGA against the GA with new selection.

		SGA		GA with new selection		
Problem Crossover		Best Average		Best	Average	Change
eil76	OX	2.79	5.70	2.58	3.19	2.51
eil76	OBX	57.81	69.89	14.13	24.23	45.66
eil76	ERX	80.86	82.03	2.79	4.03	78.00
eil76	PMX	14.87	19.14	7.25	10.84	8.30
eil76	CX	22.86	37.05	8.74	11.09	25.96
eil76	MPX	119.70	131.23	2.60	3.66	127.57
ch130	OX	7.69	10.21	7.66	9.95	0.26
ch130	OBX	185.17	202.50	70.44	90.81	111.69
ch130	ERX	220.31	238.45	6.22	8.13	230.32
ch130	PMX	55.84	56.62	9.79	11.34	45.28
ch130	CX	122.49	154.19	12.29	14.43	139.76
ch130	MPX	223.72	240.59	3.27	4.89	235.70
kroA200	OX	24.67	27.75	8.97	19.69	8.06
kroA200	OBX	362.59	389.99	100.75	175.07	214.92
kroA200	ERX	481.1	490.99	35.60	51.69	439.30
kroA200	PMX	170.51	208.11	18.47	30.23	177.88
kroA200	CX	267.98	352.01	19.82	24.46	327.54
kroA200	MPX	365.24	392.73	9.71	11.92	380.81
ftv55	OX	22.95	28.13	19.53	22.55	5.58
ftv55	OBX	32.28	50.85	12.25	37.96	12.89
ftv55	ERX	89.55	93.45	13.74	17.58	75.87
ftv55	PMX	23.07	27.07	25.57	30.67	-3.60
ftv55	CX	61.69	65.36	64.27	67.42	-2.06
ftv55	MPX	112.75	124.56	0.00	0.89	123.67

considered for the further evolutionary process when the success ratio is set to a high value. In combination with a high upper value for the maximum selection pressure genetic search can therefore be guided advantageously also for poor crossover operators, as the larger amount of handicapped offspring is simply not considered for the further evolutionary process.

For reasons of comparability of the results the parameters are set to the same values for both, the standard GA as well as for the GA with enhanced selection. The parameter settings for the test-runs shown in Tab. 2 for the standard GA as well as for the GA with the additional selection step are shown in Tab. 1.

5 Conclusion

The enhanced selection mechanism presented in this paper combines aspects of Evolution Strategies (selection pressure, success rule) and Simulated Annealing (growing selection pressure) with crossover and mutation of the general model of Genetic Algorithms. Therefore, established crossover and mutation operators for certain problems may be used analogously to the corresponding Genetic Algorithm. The investigations in this paper mainly focus on the avoidance of premature convergence and on the improvement of proven bad crossover operators.

Under special parameter settings the corresponding Genetic Algorithm is entirely included within the introduced concepts achieving an execution time only marginally worse than the execution time of the equivalent Genetic Algorithm. In other words, the introduced models can be interpreted as a generic extension of the GA-model. Therefore, an implementation of the new algorithm for a certain problem should be quite easy to do, presumed that the corresponding Genetic Algorithm (coding, operators) is known.

Especially in practical applications of Evolutionary Algorithms, where the capability of the introduced crossover operators is often not analyzed, the proposed selection should allow remarkable improvements of the solution quality. It is also believed by the authors that the use of the introduced selection should cause significant improvements when being applied to Genetic Programming applications because crossover mechanisms of Genetic Programming tend to produce offspring that do not combine the favorable properties of their parents. Furthermore, the presented selection operator should be very easily adoptable for a steady-state Genetic Algorithm and it would surely be an interesting research topic to analyze this interaction.

Based upon this newly postulated basic selection principle the mechanisms can also be combined with the already proposed Segregative Genetic Algorithm (SEGA) [3], an advanced Genetic Algorithm that introduces parallelism mainly to improve global solution quality. As a whole, a new generic evolutionary algorithm (SASEGASA) is introduced. A preliminary version of this algorithm is discussed in [5].

References

1. Affenzeller, M.: A New Approach to Evolutionary Computation: Segregative Genetic Algorithms (SEGA). Connectionist Models of Neurons, Learning Processes, and Artificial Intelligence, Lecture Notes of Computer Science 2084 (2001) 594–601
2. Affenzeller, M.: Transferring the Concept of Selective Pressure from Evolutionary Strategies to Genetic Algorithms. Proceedings of the 14th International Conference on Systems Science 2 (2001) 346–353
3. Affenzeller, M.: Segregative Genetic Algorithms (SEGA): A Hybrid Superstructure Upwards Compatible to Genetic Algorithms for Retarding Premature Convergence. International Journal of Computers, Systems and Signals (IJCSS) Vol.2 No.1 (2001) 18–32

4. Affenzeller, M.: A Generic Evolutionary Computation Approach Based Upon Genetic Algorithms and Evolution Strategies. *Journal of Systems Science* Vol.28 No.4 (2002)
5. Affenzeller, M., Wagner, S.: SASEGASA: An Evolutionary Algorithm for Retarding Premature Convergence by Self-Adaptive Selection Pressure Steering. Accepted for IWANN 2003, *Lecture Notes of Computer Science* (2003)
6. Baeck, T.: Selective Pressure in Evolutionary Algorithms: A Characterization of Selection Mechanisms. *Proceedings of the First IEEE Conference on Evolutionary Computation 1994*: (1993) 57–62
7. Goldberg, D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley Longman (1989)
8. Hajek, B.: Cooling Schedules for Optimal Annealing. *Operations Research* 13 (1988) 311–329
9. Holland, J. H.: *Adaption in Natural and Artificial Systems*. 1st MIT Press ed. (1992)
10. Larranaga, P., Kuijpers, C.M.H., Murga, R.H., Inza, I., Dizdarevic, S.: Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators. *Artificial Intelligence Review* 13 (1999) 129–170
11. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)
12. Reinelt, G.: TSPLIB – A Traveling Salesman Problem Library. *ORSA Journal on Computing* 3 (1991) 376–384
13. Schoeneburg, E., Heinzmann, F., Feddersen, S.: *Genetische Algorithmen und Evolutionsstrategien*. Addison-Wesley (1994)
14. Smith, R.E. et al.: Population Diversity in an Immune System Model: Implications for Genetic Search. *Foundations of Genetic Algorithms* 2 (1993) 153–166
15. Srinivas M. et al.: Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics* Vol.24 No.4 (1994) 656–667

Computational Methods for the Evaluation of Neuron's Firing Densities^{*}

Elvira Di Nardo¹, Amelia G. Nobile², Enrica Pirozzi³, and Luigi M. Ricciardi³

¹ Università della Basilicata,
Dipartimento di Matematica,
Potenza, Italy
`dinardo@unibas.it`

² Università di Salerno,
Dipartimento di Matematica e Informatica,
Baronissi (SA), Italy
`nobile@unisa.it`

³ Università di Napoli Federico II,
Dipartimento di Matematica e Applicazioni,
Napoli, Italy
`{enrica.pirozzi,luigi.ricciardi}@unina.it`

Abstract. Some analytical and computational methods are outlined, that are suitable to determine the upcrossing first passage time probability density for some Gauss-Markov processes that have been used to model the time course of neuron's membrane potential. In such a framework, the neuronal firing probability density is identified with that of the first passage time upcrossing of the considered process through a preassigned threshold function. In order to obtain reliable evaluations of these densities, ad hoc numerical and simulation algorithms are implemented.

1 Introduction

This contribution deals with the implementation of procedures and methods worked out in our group during the last few years in order to provide algorithmic solutions to the problem of determining the first passage time (FPT) probability density function (pdf) and its relevant statistics for continuous state-space and continuous parameter stochastic processes modeling single neuron's activity. In the neurobiological context, a classical approach to view neuronal activity as an FPT problem is to assume that a Markov process is responsible for the time course of the membrane potential under the assumption of numerous simultaneously and independently acting input processes (see also [16] and references therein). If one uses more realistic models based on correlated (non-Markov)

^{*} This work has been performed within a joint cooperation agreement between Japan Science and Technology Corporation (JST) and Università di Napoli Federico II, under partial support by INdAM (GNCS). We thank CINECA for making computational resources available to us.

Gaussian processes, serious difficulties arise because of lack of effective analytical methods for obtaining manageable closed-form expressions of the FPT pdf.

Here we shall focus on the FPT upcrossing problem that we define as an FPT problem to a boundary $S(t) \in C^1[0, +\infty)$ for the subset of sample paths of a one-dimensional non-singular Gaussian process $\{X(t), t \geq 0\}$ originating at time zero at a state X_0 , that in turn is viewed as a random variable with pdf

$$\gamma_\varepsilon(x_0) \equiv \begin{cases} \frac{f(x_0)}{P\{X(0) < S(0) - \varepsilon\}}, & x_0 < S(0) - \varepsilon \\ 0, & x_0 \geq S(0) - \varepsilon. \end{cases} \quad (1)$$

Here, $\varepsilon > 0$ is a fixed real number and $f(x_0)$ denotes the normal pdf of $X(0)$. Then,

$$T_{X_0}^{(\varepsilon)} = \inf_{t \geq 0} \{t : X(t) > S(t)\},$$

is the ε -upcrossing FPT of $X(t)$ through $S(t)$ and the related pdf is given by

$$g_u^{(\varepsilon)}(t) = \frac{\partial}{\partial t} P(T_{X_0}^{(\varepsilon)} < t) = \int_{-\infty}^{S(0)-\varepsilon} g(t|x_0) \gamma_\varepsilon(x_0) dx_0 \quad (t \geq 0), \quad (2)$$

where

$$g(t|x_0) := \frac{\partial}{\partial t} P\left(\inf_{u \geq 0} \{u : X(u) > S(u)\} < t\right), \quad X(0) = x_0 < S(0), \quad (3)$$

denotes the FPT pdf of $X(t)$ through $S(t)$.

The specific nature of various numerical methods available to compute the ε -upcrossing pdf $g_u^{(\varepsilon)}(t)$ depends on the assumptions made on $X(t)$.

2 Upcrossing FPT Densities for Gauss-Markov Processes

We recall that a non singular Gaussian process $\{X(t), t \geq 0\}$ with mean $m(t)$ is Markov if and only if its covariance is of the form

$$c(s, t) = h_1(s) h_2(t), \quad 0 \leq s \leq t < \infty, \quad (4)$$

where for $t > 0$

$$r(t) = \frac{h_1(t)}{h_2(t)} \quad (5)$$

is a monotonically increasing function and $h_1(t) h_2(t) > 0$ (cf. [8], [13]). Furthermore, any Gauss-Markov (GM) process with covariance as in (4) can be represented in terms of the standard Wiener process $\{W(t), t \geq 0\}$ as

$$X(t) = m(t) + h_2(t) W[r(t)]. \quad (6)$$

The class of all GM processes $\{X(t), t \in [0, \infty)\}$ with transition density function such that $f(x, t|y, \tau) \equiv f(x, t - \tau|y)$ is characterized by means and covariances of the following two forms:

$$m(t) = \beta_1 t + c, \quad c(s, t) = \sigma^2 s + c_1 \quad (0 \leq s \leq t < \infty, \beta_1, c \in \mathbf{R}, c_1 \geq 0, \sigma \neq 0)$$

or

$$m(t) = -\frac{\beta_1}{\beta_2} + c e^{\beta_2 t}, \quad c(s, t) = c_1 e^{\beta_2 t} \left[c_2 e^{\beta_2 s} - \frac{\sigma^2}{2c_1\beta_2} e^{-\beta_2 s} \right] \\ \left(0 \leq s \leq t < \infty, \beta_1, c, c_2 \in \mathbf{R}, \sigma \neq 0, c_1 \neq 0, \beta_2 \neq 0, c_1 c_2 - \frac{\sigma^2}{2\beta_2} \geq 0 \right).$$

The first type includes the Wiener process, while the second type includes the Ornstein–Uhlenbeck process.

For a non singular GM process with $m(t)$ and covariance $c(s, t) = h_1(s) h_2(t)$ for $s \leq t$, the pdf (1) can be immediately evaluated. Indeed, $f(x_0)$ is a normal pdf with mean $m(0)$ and variance $h_1(0)h_2(0)$ and

$$P\{X(0) < S(0) - \varepsilon\} = \frac{1}{2} \left\{ 1 + \operatorname{Erf} \left[\frac{S(0) - \varepsilon - m(0)}{\sqrt{2h_1(0)h_2(0)}} \right] \right\}. \quad (7)$$

Furthermore, the ε -upcrossing FPT pdf is the unique solution of the second kind Volterra integral equation (cf. [6]):

$$g_u^{(\varepsilon)}(t) = -2\psi_u^{(\varepsilon)}[S(t), t] + 2 \int_0^t \psi[S(t), t|S(\tau), \tau] g_u^{(\varepsilon)}(\tau) d\tau \quad (8)$$

where

$$\Psi[S(t), t|y, \tau] = \left\{ \frac{S'(t) - m'(t)}{2} - \frac{S(t) - m(t)}{2} \frac{h'_1(t)h_2(\tau) - h'_2(t)h_1(\tau)}{h_1(t)h_2(\tau) - h_2(t)h_1(\tau)} \right. \\ \left. - \frac{y - m(\tau)}{2} \frac{h'_2(t)h_1(t) - h_2(t)h'_1(t)}{h_1(t)h_2(\tau) - h_2(t)h_1(\tau)} \right\} f[S(t), t|y, \tau], \\ \psi_u^{(\varepsilon)}[S(t), t] = \int_{-\infty}^{S(0)-\varepsilon} \psi[S(t), t|x_0] \gamma_\varepsilon(x_0) dx_0.$$

A fast and accurate computational method is proposed in [6] to solve integral equation (8), that make use of the repeated Simpson rule. The proposed iteration procedure allows one to compute $\tilde{g}_u^{(\varepsilon)}(kp)$, for $k = 2, 3, \dots$, with time discretization step p in terms of computed values at the previous times $p, 2p, \dots, (k-1)p$. The noteworthy feature of this algorithm is its being implementable after simply specifying the parameter ε , functions $m(t), h_1(t), h_2(t)$ that characterize the process, boundary $S(t)$ and discretization step p . Furthermore, it does not involve any heavy computation, neither it requires use of any library subroutines, Monte Carlo methods or other special software packages to calculate high dimension multiple integrals.

3 Kostyukov Model

In the context of single neuron's activity modeling a completely different, apparently not well known, approach was proposed by Kostyukov *et al.* in [10] and

[11] in which a non-Markov process of a Gaussian type is assumed to describe the time course of the neural membrane potential.

Kostyukov model (K-model) makes use of the notion of correlation time. Namely, let $X(t)$ be a stationary Gaussian process with zero mean, unit variance and correlation function $R(t)$. Then,

$$\vartheta = \int_0^{+\infty} |R(\tau)| d\tau < +\infty$$

is defined as the correlation time of the process $X(t)$. Under the assumption that $\lim_{\varepsilon \rightarrow 0} P\{X(0) < S(0) - \varepsilon\} \simeq 1$, i.e. $\lim_{\varepsilon \rightarrow 0} \gamma_\varepsilon(x_0) \simeq f(x_0)$, Kostyukov works out an approximation $q(t)$ to the upcrossing FPT pdf. This approximation is obtained as solution of the integral equation

$$\int_0^t q(\tau) K(t, \tau) d\tau = 1 - \Phi[S(t)], \quad (9)$$

where

$$K(t, \tau) = \begin{cases} \frac{1}{2}, & t = \tau \\ 1 - \Phi \left\{ \frac{(t - \tau + \vartheta) S(t) - \vartheta S(\tau)}{\sqrt{(t - \tau + \vartheta)(t - \tau)}} \right\}, & t > \tau, \end{cases}$$

and where $\Phi(z)$ is the distribution function of a standard Gauss random variable. Note that equation (9) can be solved by routine methods. Furthermore, under the above approximation, in equation (9) the unique parameter ϑ characterizes the considered class of stationary standard Gaussian processes.

4 Upcrossing FPT Densities for Stationary Gaussian Processes

Let $X(t)$ be a stationary Gaussian process with mean $m(t) = 0$ and covariance $E[X(t)X(\tau)] = c(t, \tau) = c(t - \tau)$ such that $c(0) = 1$, $\dot{c}(0) = 0$ and $\ddot{c}(0) < 0$. By using a straightforward variant of a method proposed by Ricciardi and Sato ([14], [15]) for the determination of the conditional FPT density g , in [7] we have obtained the following series expansion for the upcrossing FPT pdf:

$$g_u^{(\varepsilon)}(t) = W_1^{(u)}(t) + \sum_{i=1}^{\infty} (-1)^i \int_0^t dt_1 \int_{t_1}^t dt_2 \cdots \int_{t_{i-1}}^t dt_i W_{i+1}^{(u)}(t_1, \dots, t_i, t). \quad (10)$$

Here,

$$W_{i+1}^{(u)}(t_1, \dots, t_i, t) = \left[\int_{-\infty}^{S(0)-\varepsilon} f(z) dz \right]^{-1} \int_{-\infty}^{S(0)-\varepsilon} W_{i+1}(t_1, \dots, t_i, t | x_0) f(x_0) dx_0, \quad (11)$$

where $W_n(t_1, \dots, t_n | x_0) dt_1 \cdots dt_n$, $\forall n \in \mathbf{N}$ and $0 < t_1 < \dots < t_n$, denotes the joint probability that $X(t)$ crosses $S(t)$ from below in the time intervals $(t_1, t_1 + dt_1), \dots, (t_n, t_n + dt_n)$ given that $X(0) = x_0$.

The evaluation of the partial sums of the above series expansion is made hardly possible because of the outrageous complexity of the functions $W_n^{(u)}$ and of their integrals. However, approximations of upcrossing FPT density can be carried out by evaluating first of all $W_1^{(u)}(t)$. The explicit expression of $W_1^{(u)}(t)$ is (cf. [4]):

$$\begin{aligned} W_1^{(u)}(t) = & \frac{\exp\left\{-\frac{S^2(t)}{2}\right\}}{2\pi \left[1 + \operatorname{Erf}\left(\frac{S(0) - \varepsilon}{\sqrt{2}}\right)\right]} \left\{ [-\ddot{c}(0)]^{1/2} \exp\left(-\frac{[\dot{S}(t)]^2}{2[-\ddot{c}(0)]}\right) [1 + \operatorname{Erf}(U_\varepsilon(t))] \right. \\ & - \frac{\dot{c}(t)}{\sqrt{1 - c^2(t)}} \exp\left(-\frac{[S(0) - \varepsilon - S(t)c(t)]^2}{2[1 - c^2(t)]}\right) [1 - \operatorname{Erf}(V_\varepsilon(t))] \\ & \left. - \frac{\dot{S}(t)}{\sqrt{1 - c^2(t)}} \int_{-\infty}^{S(0) - \varepsilon} \exp\left(-\frac{[x_0 - S(t)c(t)]^2}{2[1 - c^2(t)]}\right) \left[1 - \operatorname{Erf}\left(\frac{\sigma(t|x_0)}{\sqrt{2}}\right)\right] dx_0 \right\} \end{aligned} \quad (12)$$

where:

$$\begin{aligned} A_3 &= \begin{pmatrix} 1 & c(t) & \dot{c}(t) \\ c(t) & 1 & 0 \\ \dot{c}(t) & 0 & -\ddot{c}(0) \end{pmatrix} \\ \sigma(t|x_0) &= \left(\frac{1 - c^2(t)}{\|A_3\|}\right)^{1/2} \left\{ \dot{S}(t) + \frac{\dot{c}(t)[c(t)S(t) - x_0]}{1 - c^2(t)} \right\} \\ \operatorname{Erf}(z) &:= \frac{2}{\sqrt{\pi}} \int_0^z e^{-y^2} dy \\ U_\varepsilon(t) &:= \frac{-\ddot{c}(0)[S(0) - \varepsilon - S(t)c(t)] - \dot{c}(t)\dot{S}(t)}{\sqrt{2} \|A_3\| [-\ddot{c}(0)]} \\ V_\varepsilon(t) &:= \frac{-\dot{c}(t)[S(0) - \varepsilon - S(t)c(t)] + \dot{S}(t)[1 - c^2(t)]}{\sqrt{2} \|A_3\| [1 - c^2(t)]}. \end{aligned}$$

A numerical approximation of (12) was proposed in [4] and evaluated by using NAG routines based on an adaptive procedure described in [1]. For each $t > 0$, the function $W_1^{(u)}(t)$ provides an upper bound to the upcrossing FPT pdf. The numerical computations indicate that this can be taken as a good approximation of $\tilde{g}_u^{(\varepsilon)}(t)$ only for small values of t .

Equations (10) and (11) call for alternative procedures to gain more information on upcrossing FPT pdf. To this aim, we have updated an algorithm (cf. [2]) for the construction of sample paths of the specified stationary Gaussian process $X(t)$, with random initial point, under the assumption of rational spectral density such that the degree of the polynomial in its denominator is larger

than that in the numerator. Since the sample paths of the simulated process are generated independently of each other, the simulation procedure is particularly suited to run on supercomputers. A parallel simulation procedure has been implemented on a IBM SP-Power4 machine to generate the sample paths of $X(t)$ and to record their upcrossing FPT times through the preassigned boundary in order to construct reliable histograms estimating the FPT pdf $\tilde{g}_u^{(\varepsilon)}(t)$. To evaluate the upcrossing FPT densities, we have chosen X_0 randomly according to the initial pdf $\gamma_\varepsilon(x_0)$. To this purpose, we have made use of the following acceptance-rejection method (cf. for instance [12]):

STEP 1 Generation of pseudo-random numbers U_1, U_2 uniformly distributed in $(0, 1)$;

STEP 2 $Y \leftarrow \log U_2 + S(0) - \varepsilon$;

STEP 3 if $U_1 < \exp\left\{-\frac{(Y+1)^2}{2}\right\}$ then $X_0 \leftarrow Y$ else goto STEP 1;

STEP 4 STOP.

Let us observe that the random variable Y in STEP 2 is characterized by the pdf

$$h(y) = \begin{cases} e^{y-[S(0)-\varepsilon]}, & \text{if } y < S(0) - \varepsilon \\ 0, & \text{if } y \geq S(0) - \varepsilon. \end{cases} \quad (13)$$

We point out that our simulation algorithm stems directly out of Franklin's algorithm [9]. We have implemented it in both vector and parallel modalities (see [2], [5]) after suitably modifying it for our computational needs: Namely, to obtain reliable approximations of upcrossing densities (cf. [3], [4], [5]). Thus doing, reliable histograms of FPT densities of stationary Gaussian processes with rational spectral densities can be obtained in the presence of various types of boundaries.

5 Computational Results

In order to compare the results obtained via different methods for determining the upcrossing FPT pdf, we start considering the particular stationary standard Gaussian process $X(t)$ having correlation function

$$R(t) = e^{-\beta|t|} \cos(\alpha t), \quad (14)$$

where $\alpha = 10^{-5}$ and $\beta = \vartheta^{-1}$. The approximation $\tilde{g}_u(t)$ for the FPT density of this process in the presence of the threshold

$$S(t) = -t^2/2 - t + 5 \quad (15)$$

is estimated via 10^6 simulated paths with the following choices of correlation times:

$$(i) \quad \vartheta = 0.008, \quad 0.016, \quad 0.032, \quad 0.064, \quad 0.128, \quad 0.256, \quad 0.512, \quad 1.024,$$

in Fig. 1(a) and with

$$(ii) \quad \vartheta = 2.048, \quad 4, \quad 8, \quad 16, \quad 32, \quad 64, \quad 100, \quad 200,$$

in Fig. 1(b).

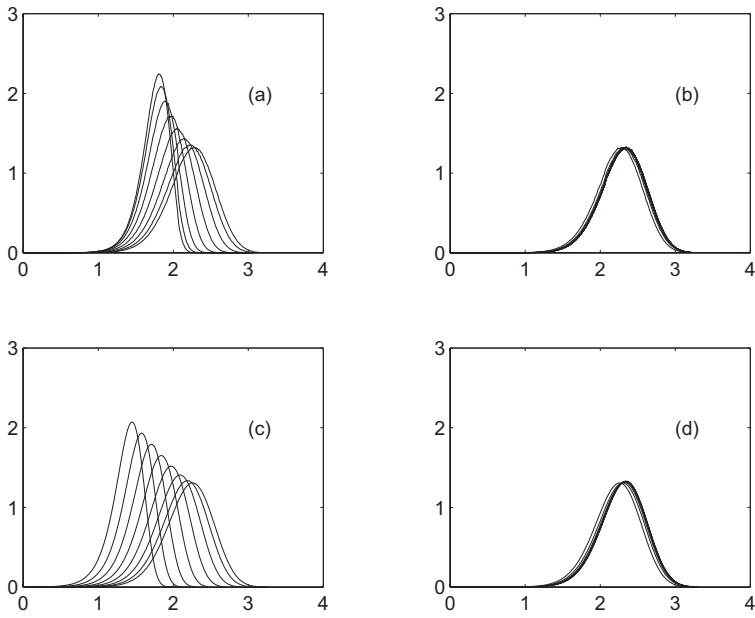


Fig. 1. Plot of the simulated $\tilde{g}_u(t)$ in Fig. 1(a) and in Fig. 1(b) and plot of $\tilde{g}_u(t)$ for the OU-model in Fig. 1(c) and in Fig. 1(d), with threshold $S(t) = -t^2/2 - t + 5$

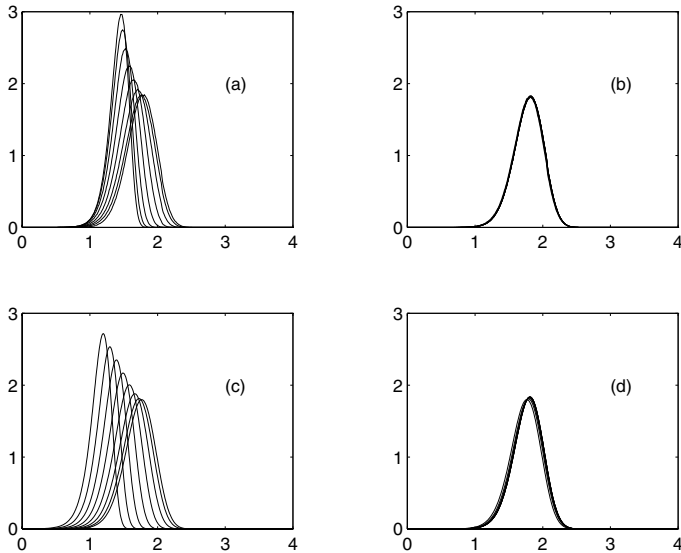


Fig. 2. Plot of the simulated $\tilde{g}_u(t)$ in Fig. 2(a) and in Fig. 2(b) and plot of $\tilde{g}_u(t)$ for the OU-model in Fig. 2(c) and in Fig. 2(d), with threshold $S(t) = -t^2 - t + 5$

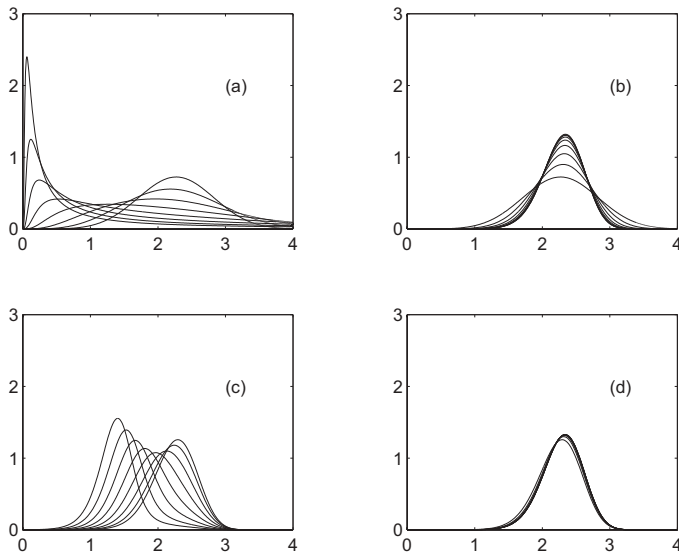


Fig. 3. Plot of $\tilde{g}_u(t)$ for the Wiener model in Fig. 3(a) and in Fig. 3(b) and plot of $q(t)$ for the Kostyukov-model in Fig. 3(c) and in Fig. 3(d), with threshold $S(t) = -t^2/2 - t + 5$

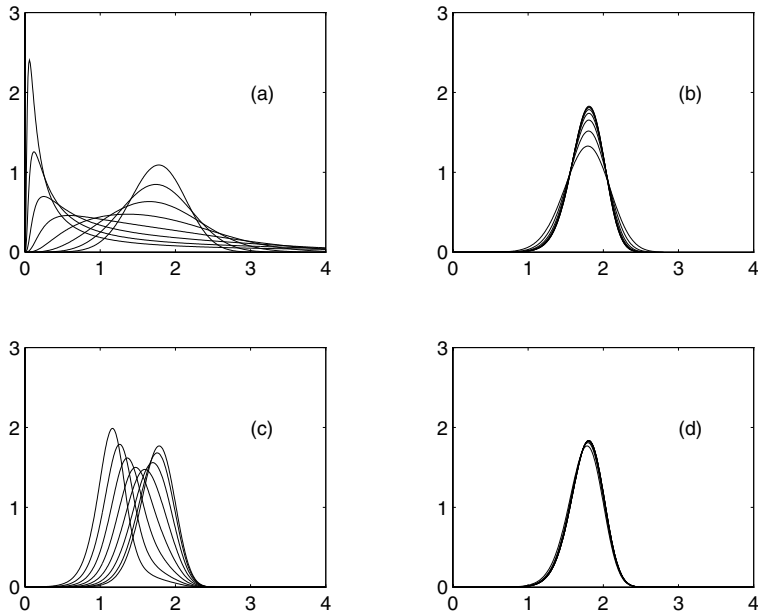


Fig. 4. Plot of $\tilde{g}_u(t)$ for the Wiener model in Fig. 4(a) and in Fig. 4(b) and plot of $q(t)$ for the Kostyukov-model in Fig. 4(c) and in Fig. 4(d), with threshold $S(t) = -t^2 - t + 5$

Furthermore, for the stationary GM process $X(t)$ with

$$m(t) = 0, \quad c(s, t) = e^{-(t-s)/\vartheta} \quad (s < t), \quad (16)$$

known as the Ornstein-Uhlenbeck (OU) model, the FPT density approximation $\tilde{g}_u(t)$ in the presence of threshold (15) is evaluated via (8) and is plotted in Fig. 1(c) and Fig. 1(d) for the values of ϑ respectively indicated in (i) and (ii). Note that for values of α close to zero, (14) is near the OU correlation function, given by

$$e^{-t/\vartheta} \quad (t \geq 0).$$

Figure 2 is the same of Fig. 1 for the threshold

$$S(t) = -t^2 - t + 5. \quad (17)$$

We notice that for large correlation times the firing densities in Fig. 1(b) exhibit features similar to those of the OU-model in Fig. 1(d). Similar considerations hold for the case of Fig. 2(b) and Fig. 2(d).

Let us now focus our attention on a non stationary GM process $X(t)$ (Wiener-model) with

$$m(t) = 0, \quad c(s, t) = s/\vartheta \quad (s < t). \quad (18)$$

The approximation $\tilde{g}_u(t)$ for this process in the presence of threshold (15) is estimated via (8) for the choices of the ϑ as (i) in Fig. 3(a) and as (ii) in Fig. 3(b). Finally, the function $q(t)$ of the K-model in the presence of threshold (15) is evaluated via (9) with ϑ as (i) in Fig. 3(c) and as (ii) in Fig. 3(d). Figure 4 is the same as Fig. 3 but for threshold (17).

We point out that again for large values of ϑ the firing densities in Fig. 3(b) exhibit features similar to those of the Wiener-model in Fig. 3(d). The same occurs in Fig. 4(b) and Fig. 4(d). Hence, the validity of approximations of the firing densities in the presence of memory effects by the FPT densities of Markov type is clearly related to the magnitude of the involved correlation time. Indeed, it could be shown that the asymptotic behavior of all these models becomes increasingly similar as ϑ grows larger.

References

1. De Donker, E.: An adaptive extrapolation algorithm for automatic integration Signum Newsletter, Vol. 13, 2, (1978) 12–18
2. Di Nardo, E., Pirozzi, E., Ricciardi, L.M., Rinaldi, S.: Vectorized simulations of normal processes for first crossing-time problems. Lecture Notes in Computer Science, Vol. 1333, (1997) 177–188
3. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: On a non-Markov neuronal model and its approximation. BioSystems, Vol. 48, (1998) 29–35
4. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: Evaluation of upcrossing first passage time densities for Gaussian processes via a simulation procedure. Atti della Conferenza Annuale della Italian Society for Computer Simulation. (1999) 95–102

5. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: Simulation of Gaussian processes and first passage time densities evaluation. *Lecture Notes in Computer Science*, Vol. 1798, (2000) 319–333
6. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: A computational approach to first-passage-time problem for Gauss-Markov processes. *Adv. Appl. Prob.*, Vol. 33, (2001) 453–482
7. Di Nardo, E., Nobile, A.G., Pirozzi, E., Ricciardi, L.M.: Gaussian processes and neuronal models: an asymptotic analysis. *Cybernetics and Systems*, Vol. 2, (2002) 313–318
8. Doob, J.L.: Heuristic approach to the Kolmogorov-Smirnov theorem. *Ann. Math. Statist.*, Vol. 20, (1949) 393–403
9. Franklin, J.N.: Numerical simulation of stationary and non stationary gaussian random processes. *SIAM Review*, Vol. 7, (1965) 68–80
10. Kostyukov, A.I.: Curve-Crossing Problem for Gaussian Stochastic Processes and its Application to Neural Modeling. *Biol. Cybernetics*, Vol. 29, (1978) 187–191
11. Kostyukov, A. I., Ivanov, Yu.N. and Kryzhanovsky, M.V.: Probability of Neuronal Spike Initiation as a Curve-Crossing Problem for Gaussian Stochastic Processes. *Biological Cybernetics*, Vol. 39, (1981) 157–163
12. Knuth, D. E.: The art of computer programming, Vol. 2, Reading, M.A. Addison Welsey, (1973)
13. Mehr, C.B. and McFadden, J.A.: Certain Properties of Gaussian Processes and their First-Passage Times. *J. R. Statist. Soc. (B)*, Vol. 27, (1965) 505–522
14. Ricciardi, L.M. and Sato, S.: A note on first passage time for Gaussian processes and varying boundaries. *IEEE Trans. Inf. Theory*, Vol. 29, (1983) 454–457
15. Ricciardi, L.M. and Sato, S.: On the evaluation of first-passage-time densities for Gaussian processes. *Signal Processing*, Vol. 11, (1986) 339–357
16. Ricciardi, L.M. and Lánský, P.: Diffusion models of neuron activity. In: *The Handbook of Brain Theory and Neural Networks* (M.A. Arbib, ed.). The MIT Press, Cambridge. (2002) 343–348

Developing the Use of Process Algebra in the Derivation and Analysis of Mathematical Models of Infectious Disease

R. Norman and C. Shankland

Department of Computing Science and Mathematics,
University of Stirling, UK.
{ces,ran}@cs.stir.ac.uk

Abstract. We introduce a series of descriptions of disease spread using the process algebra WSCCS and compare the derived mean field equations with the traditional ordinary differential equation model. Even the preliminary work presented here brings to light interesting theoretical questions about the “best” way to defined the model.

1 Motivation

Moving from individual to population level processes is a key challenge of biological theory. Within the life sciences community there is an increasing need to model various systems at a level where individuals (whether cells or whole organisms) interact and from this to be able to determine the behaviour of a group or population of those individuals. Individual variation is likely to be important in, for example, predicting disease spread, explaining social insect behaviour or predicting the effect of new drugs at the molecular level. Systems of non-linear coupled differential equations are often used to describe the spread of infectious diseases from a population level [1]. Mathematical techniques are available to analyse algebraically the long-term behaviour of these models. However, once the level of realism in these models increases the mathematical techniques currently in use often cannot give a full picture of the system dynamics. Further, this approach often assumes population level behaviour is understood without taking individual behaviour, and hence variation, into account. Developing an analytical framework for individual based models is necessary if we are to understand the contribution of the individual to population level behaviours. This is particularly important in situations where the observation of the biological system takes place at the level of the individual.

One commonly used modelling approach that explicitly simulates a collection of individuals is the use of probabilistic cellular automata [2,3]. These are often grid-based models in which each cell within the grid consists of an individual that moves and interacts with its neighbours according to a strict set of rules. If we then look at the individuals summed across the whole grid we can infer population level behaviour. This method does, therefore, fit the desired biological criteria stated above. However, the method does have some limitations as it is

usually simulation based with no algebraic analysis possible. Although simulation is useful, it is not computationally feasible to explore the whole parameter space using simulation therefore there is a risk that some important behaviour has been missed. Recent innovative work on cellular automata has allowed some algebraic progress to be made. The method of pair approximations has been used to allow some rules of general behaviour of the system to be determined [4]; however, these *approximate* the *spatial* relationships within a population and do not include the individual processes explicitly.

A method of modelling biological problems which is gaining increasing recognition is the application of theoretical computing science techniques, such as Process Algebra. Process algebra allows description of the rules of behaviour and interaction for an individual. Individuals are loosely coupled together to describe a system. Typically, process algebras are used to describe concurrent distributed computer systems, such as communications protocols, but some have already been applied to biological systems. For example, the discrete time process algebra Weighted Synchronous Calculus of Communicating Systems (WSCCS) [5] has been useful in describing dynamics of social insect populations [6,7], supported by use of the tool Probability Workbench (PW). Process algebra allows creation of an individual-based model, but with the advantage of a formal mathematical semantics, thereby allowing formal analysis. Both long and short term behaviours of the population as a whole may be rigorously explored by examining the Markov chain resulting from the description, or by derivation of a mean field model (MFE) [7, Chapter 3]. The tool PW supports these activities to some extent, as well as providing simulation facilities, which are useful for initial confirmation that the model behaves as expected. The fact that several different and powerful mathematical analyses are possible means that process algebra has a huge potential for allowing novel and innovative approaches to the modelling of biological systems.

The question of how those process algebra models relate to the traditional population level models has not yet been fully explored. In this paper we will concentrate on the use of process algebra, WSCCS in particular, to model systems of disease spread. We carry out numerical analyses of these systems using the Probability Workbench and manual probabilistic analysis. We compare the standard Ordinary Differential Equations (ODEs) to mean field approximations for population level behaviour derived from the WSCCS descriptions. This allows us to better understand the correlation between individual and population level behaviour. Simply concentrating on the average behaviour of these systems can answer important biological questions and bring to light interesting questions about using process algebra to express biological systems. By beginning with such simple, well-understood systems we also gain a better understanding of the relationship between the rules of individual behaviour as expressed in process algebra, and the derived MFEs. A particular biological problem we plan to address in future work is described in Section 4, and our conclusions are made in Section 5.

2 Background

2.1 Population Level Model

Consider a system of disease spread in which the population is split into three groups of individuals: susceptible (S), infected (I), and recovered (R). In this simplest case we assume no births or deaths, and no deaths due to the disease. We are therefore considering an epidemic in a closed population in which the only things that can happen are transmission of the disease from I to S at rate β , and recovery from I to R at rate γ . The most commonly used discrete time form of the equations which describe the system dynamics are as follows:

$$\begin{aligned} S_{t+1} &= S_t - \beta S_t I_t \\ I_{t+1} &= I_t + \beta S_t I_t - \gamma I_t \\ R_{t+1} &= R_t + \gamma I_t \end{aligned} \tag{1}$$

See the biological problem presented in Section 4 for more detail on this particular formalisation. This model will be used as a reference point for the process algebra descriptions of Section 3.

2.2 WSCCS Notation

For a full introduction to the notation of WSCCS see Tofts' introductory paper [5]. We give here an overview of the notation, using excerpts of the examples of Section 3. The biological implications of these will be discussed in full in the next section. The notation used is the ASCII variant of WSCCS accepted by the tool Probability Workbench. descriptions directly for themselves.

As with other process algebras, *agents* describe behaviours. Agents are composed using the key operations of action prefix, choice, and parallelism. Actions are named events that we wish to observe, for example, “send a message”, “receive a message”, “toss a coin and get heads”, “toss a coin and get tails”, “pass on an infection”, and so on. The special action *tick*, written $\mathfrak{t}$, simply allows time to pass. There is no notion of real time in WSCCS, but there is a notion of ordering. For example, the following PW input describes an agent **S2**:

```
bs S2 1.infect:I1 + 1.t:S1
```

S2 can perform the action **infect** and subsequently behave like **I1** (i.e. this individual has become infected), or **S2** can perform the action **t** and subsequently behave like **S1** (i.e. this individual stays susceptible). **bs** is an instruction to PW to direct it to interpret what follows as a basic sequential agent definition.

Since WSCCS is a probabilistic process algebra, the choice can be affected by *weights*. Above, the weights before the actions are both 1, meaning that each branch is equally likely (although this can be modified by communication between agents, see below). In the descriptions of Section 3 weights less than 1 are used, adding up to 1 across a choice, so weights can be thought of as probabilities.

For example, the following PW input describes an agent **I1**:

bs I1 pr.t:R2 + pa.t:T2 + (1-pr-pa).t:I2

I1 can evolve in three ways. With probability **pr**, a tick action occurs and I1 subsequently behaves like the agent R2. With probability **pa**, a tick action occurs and I1 subsequently behaves like the agent T2. Finally, with probability **(1-pr-pa)**, a tick action occurs and I1 subsequently behaves like the agent I2.

More interesting behaviour arises when a number of agents are composed in parallel, allowing interaction between agents. The following expression introduces an agent **W2**:

bpa W2 I2|Trans

Agent **W2** is composed of the two agents **I2** and **Trans** running in parallel. The term **bpa** is an instruction to PW that this is a basic parallel process.

Communication between agents arises when an action and its *inverse* occur:

bs Trans 1.infect⁻¹:I1 + 1.t:I1

The **infect⁻¹** action is a partner to the **infect** action of **S2** above. The composition of **infect** and **infect⁻¹** is a tick action. Communication is strictly one to one.

The following expression introduces an agent **Population**:

btr Population S1|S1|S1|S1|I1/L

Population is composed of four instances of the agent **S1** and one **I1**. The **/L** at the end is *restriction*. This means that only actions in the set **L** are permitted to occur. In the models of Section 3 **L** is defined to contain only the action **t**. By only allowing tick actions to occur we disallow solitary **infect** and **infect⁻¹** actions. In Section 3 only the agents are described explicitly. To complete each model a parallel expression describing **Population** as above is required. The term **btr** is an instruction to PW that this is a agent consisting of processes in parallel, with restriction, and *priority* occurring.

Priority behaves like an infinite weight on actions. For example:

bs T2 1@1.infect⁻¹:I1 + 1.t:I1

Here the communication action has priority over the tick action (signified by the **1@**). Practically, this means that if there is another process with which to communicate, then **T2** must perform the **infect⁻¹** action. The alternative branch can only be taken if there are no other processes with which to communicate.

The features of WSCCS are common to many probabilistic process algebras. We have chosen to work initially with WSCCS because of its historical association with biological problems, and the availability of tool support.

2.3 Overall Picture of Our Research Plan

Our long-term research programme addresses three open questions: Can we improve the mathematical models which describe infectious diseases by comparing

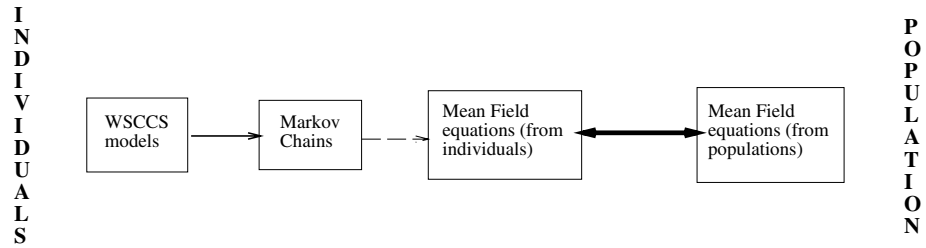


Fig. 1. The Relationship between the Process Algebra and PDE Approach

and combining the individual and population approaches? Can process algebras yield more tractable analyses than the differential equation based models for realistic models of disease spread? How does the ability to model explicit rules of behaviour and interaction of individuals, permitted by the process algebra, affect model results? This paper describes our initial investigations into these questions.

Our approach is illustrated in Figure 1. The focus of our work is developing methods of deriving accurate Mean Field Equations (MFEs) from the process algebra (the dashed arrow), the comparison of the two sets of MFEs obtained (the thick double headed arrow), and the comparison of the results with real world data.

Sumpter [7, Section 3.4] described a heuristic for deriving MFEs from the process algebra specification which we adopt here. We expect that, in general, more realistic rules of individual behaviour will result in different population level models to those traditionally used. This may additionally enhance our understanding of disease systems, and allow us to reformulate the assumptions made in modelling at the population level.

3 Individual Based Models in WSCCS

We will now describe the development of an individual based model using WSCCS which gives mean field equations equivalent to those derived at the population level (1). We start with an existing model [7], and then develop it to increase its biological realism and to make its MFE more similar to the traditional model (1). At this point we are refining our method and do not expect that the individual based model should be different from the population based model. At each stage interesting theoretical and biological questions arise.

In the initial model [7, Section 3.4] agents are either infected, susceptible or infected. The WSCCS description of the model is in Figure 2. Here, as in all the models, there are two steps. First the choice to either recover, actively try to infect, or do nothing is taken. Second the actual communication which produces infection is made. This separation of concerns is essential. A naive model might have both of these in one step; however, the communication mechanism

of WSSCS may distort the effect of the weights on the choice, thereby giving an incorrect model.

In Figure 2 at each tick an infected individual recovers with probability p_r , is infectious with probability p_a , or simply remains infected with probability $1 - p_r - p_a$. Susceptible individuals are available to be infected with probability p_c and remain susceptible with probability $1 - p_c$. The use of priorities on the infect actions means that the maximum number of contacts will occur between those agents who are available to infect ($p_a I_t$, where I_t is the number of I1 processes at time t) and those who are available to be infected ($p_c S_t$, where S_t is the number of S1 processes at time t). Therefore, after two ticks (one time step in the MFE) there will be n new infected individuals, where n is the minimum of $p_a I_t$ and $p_c S_t$.

```

bs I1
pr.t:R + pa.t:T2 + (1-pr-pa).t:I2
bs I2 1.t:I1
bs T2 1@1.infect^-1:I1 +
1.t:I1
bs S1 pc.t:P2 + (1-pc).t:S2
bs S2 1.t:S1
bs P2 1@1.infect:I1 +
1.t:S1
bs R1 1.t:R2
bs R2 1.t:R1
    
```

Fig. 2. Version 1: Sumpter 2000: some susceptibles available for infection

In this case the mean field equations are as follows. We use the method of derivation described by Sumpter in his thesis [7, Chapter 3].

$$\begin{aligned}
 S_{t+1} &= (1 - p_c)S_t + \max\{p_c S_t - p_a I_t, 0\} \\
 I_{t+1} &= (1 - p_r)I_t + \min\{p_a I_t, p_c S_t\} \\
 R_{t+1} &= R_t + p_r I_t
 \end{aligned} \tag{2}$$

These are clearly not in the same form as the population level equations (1) because of the maximum and minimum terms. In addition, the model of Figure 2 is not very realistic biologically since it is unlikely that susceptibles will independently become available for infection.

Our first attempt to make the model more realistic was to make *all* susceptibles available for infection every time step. Clearly this means a change in interpreting the role of p_a , since now this is the only probability contributing to infection. The WSSCS notation for this model is given in Figure 3. The probability p_r is as described for the previous case.

The mean field equations are as follows:

$$\begin{aligned}
 S_{t+1} &= S_t - \min\{S_t, p_a I_t\} \\
 I_{t+1} &= (1 - p_r)I_t + \min\{p_a I_t, S_t\} \\
 R_{t+1} &= R_t + p_r I_t
 \end{aligned} \tag{3}$$


```

bs I1 pr.t:R2 + pa.t:T2 +
(1-pr-pa).t:I2
bs I2 1.t:I1
bs T2 1@1.infect~-1:I1 + 1.t:I1
bs S1 1.t:S2
bs
S2 1@1.infect:I1 + 1.t:S1
bs R1 1.t:R2
bs R2
1.t:R1

```

Fig. 3. Version 2: all susceptibles available for infection

Depending on the value selected for p_a and the number of susceptibles and infecteds, the system evolves by infecting exactly $p_a I_t$ at each step, until the final step when the *min* terms become relevant and the last few susceptibles are infected. However, this model is very unsatisfactory biologically since many infections are guaranteed every time step, whereas in reality, as more individuals in the population become infected and recovered, it will be more difficult to find susceptibles and so infection should become more unlikely.

Biologically, we therefore require some rules that mean that infectious individuals can “waste” the chance to infect by communicating with infected or recovered individuals in a given time step as well as susceptibles. Our third model is given in Figure 4.

```

bs I1 pr.t:R2 + pa.t:T2
+ (1-pr-pa).t:I2
bs I2 1@1.infect:I1 + 1.t:I1
bs T2 1@1.infect~-1:I1 +
1.t:I1
bs S1 1.t:S2
bs S2 1@1.infect:I1 + 1.t:S1
bs R1 1.t:R2
bs R2
1@1.infect:R1 + 1.t:R1

```

Fig. 4. Version 3: allowing infection attempts to be “wasted” on infecteds and recovered

The ability to communicate on the **infect** action has been added to both I2 and R2 processes. This means that, as desired, there is now some probability that when a T2 process makes its **infect~-1** action, the agent it is communicating with will not be a susceptible, and no new infection will result.

The mean field equations are:

$$\begin{aligned}
 S_{t+1} &= S_t - (p_a I_t S_t) / (S_t + (1 - p_a) I_t + R_t) \\
 I_{t+1} &= (1 - p_r) I_t + (p_a I_t S_t) / (S_t + (1 - p_a) I_t + R_t) \\
 R_{t+1} &= R_t + p_r I_t
 \end{aligned} \tag{4}$$

An obvious problem with this model is that although infections can be passed to more of the population than in the model of Figure 3, one class is missed out from the communication, that of the transmitting infecteds (T2 processes).

There are at least two ways to add communication between infectious individuals to this model using WSCCS. Both sets of notation give the same simulation results and the same mean field equations; however, what is interesting is whether there are other theoretical implications to the representation, what the advantages and disadvantages of each notation are, and whether one is a more efficient representation for calculation, for example in PW, than the other. Other differences may appear when other forms of analysis, such as Markov chain analysis, are carried out, but this is beyond the scope of the present work.

In the fourth version, given in Figure 5, all individuals may communicate with an infectious individual, regardless of whether they are susceptible, already infected, or recovered (and therefore immune). In particular, we want to express that even other infectious individuals can communicate with each other. The mechanism for this is slightly clumsy. An infectious individual is represented by a pair of processes in parallel: one to do the infect^{-1} action (I2), and the other to possibly do the infect action (Trans).

Note that this merely presents an additional opportunity to “waste” an infection; we do not enforce that the **Trans** process *has* to communicate with the particular I2 agent it has been paired with. Recall that this agent will also run in parallel with all the other individuals in the systems, therefore it could easily communicate with any of the others, including the susceptibles. This notation introduces an extra agent **Trans** into the system. These are removed (by becoming the null agent **T** which is automatically absorbed) after a single infect^{-1} or tick action. They are not counted twice. This notation, given in Figure 5, is more explicit since we are determining what happens to each individual.

```

bs I1 pr.t:R2 + pa.t:T2 +
(1-pr-pa).t:I2
bs I2 1@1.infect:I1 + 1.t:I1
bpa T2 I2|Trans
bs Trans
1@1.infect-1:T + 1.t:T
bs S1 1.t:S2
bs S2 1@1.infect:I1 + 1.t:S1
bs R1
1.t:R2
bs R2 1@1.infect:R1 + 1.t:R1

```

Fig. 5. Version 4: Explicit notation: forced communication, all available to infect

The alternative way to describe this is by taking the model of Figure 4 and relinquishing the use of priority in communications. This exploits the way that actions occur in WSCCS and the underlying semantics of the language. The notation, seen in Figure 6, is somewhat more concise and elegant. Instead of forcing communication to occur wherever possible, we allow an equally weighted choice

```

bs I1 pr.t:R2 + pa.t:T2 +
(1-pr-pa).t:I2
bs I2 1.infect:I1 + 1.t:I1
bs T2 1.infect^-1:I1 + 1.t:I1
bs
S1 1.t:S2
bs S2 1.infect:I1 + 1.t:S1
bs R1 1.t:R2
bs R2 1.infect:R1 +
1.t:R1

```

Fig. 6. Version 5: Implicit notation: all available to infect, except T2

between communicating and not communicating. As above, new infections are not guaranteed, and depend on p_a , and the number of S2, I2, and R2 processes.

As stated above the two sets of notation give the same simulation results. The first one contains more agents (although temporary) which may be important when models become more complicated, but is more explicit and the probabilistic rules of interaction are clearer in this case. The second set of notation relies more on the underlying rules of WSCCS and may be more efficient both to write down and to simulate but relies on a deeper understanding of WSCCS and so may be more open to errors, particularly if non-experts start to use these methods.

Given both sets of notation above the mean field approximation is as follows:

$$\begin{aligned}
 S_{t+1} &= S_t - p_a I_t S_t / (S_t + I_t + R_t) \\
 I_{t+1} &= (1 - p_r) I_t + p_a I_t S_t / (S_t + I_t + R_t) \\
 R_{t+1} &= R_t + p_r I_t
 \end{aligned} \tag{5}$$

We have now lost the minimum and maximum terms and this model is now equivalent to the population level model (1) with $\gamma = p_r$ and $\beta = p_a / (S_t + I_t + R_t)$ since $S_t + I_t + R_t$ is constant in this model.

This is an important breakthrough in the development of these models since we have now made the connection between the population models and the individual-based models. This allows us a strong base from which to start addressing other important biological questions.

4 Future Directions: A Biological Problem

Models of infectious disease spread have been used successfully for a number of years to predict the dynamics and control of a variety of both human and animal diseases. In particular they have been used increasingly to determine government policy in the control of diseases such as BSE, and Foot and Mouth disease. A large body of work in this area has built up, particularly since the ground breaking work of Anderson and May in the early 1980s [1]. However, the majority of this work is based on population level behaviour and assumes a priori that we know what is happening to the population as a whole.

One common assumption of these models that has come under question recently is the way in which the spread of infection is modelled. The majority of these models assume that the rate at which susceptible individuals become infected is proportional to the density of susceptible and infectious individuals and is written as βSI where S is the density of susceptibles, I is the density of infectious individuals and β is the transmission rate. When transmission is written in this form it is known as *density dependent transmission*.

An alternative term which is beginning to be used is *frequency dependent transmission*. This assumes that the number of contacts an individual has per day is independent of the density of individuals around it, and has traditionally been used to describe sexually transmitted diseases or vector borne diseases. There is however an increasing body of evidence that it might be more appropriate than density dependent transmission in systems in which animals live in social groups and mainly interact with their own groupings, for example, rabbits in a warren. Frequency dependent transmission appears in the equations as the term $\beta' SI/N$ where N is the total population density.

As stated above, these transmission terms assume that we know what is happening at the population level whereas in reality it is more likely that, experimentally, we have observations on how individuals interact. We are interested, firstly in what rules of individual behaviour would give rise to density or frequency dependent transmission terms at the population level, and ultimately what realistic rules of individual behaviour would mean in terms of population level transmission.

In this paper we have laid the ground work to allow these questions to be investigated. Here, we examined a simple disease system in which the population is closed, i.e. there are no births and deaths. It should be noted that in a closed population, density and frequency dependent transmission are just scalings of one another since N is constant. The WSCCS descriptions of Figures 5 and 6 seem to capture frequency dependent transmission.

5 Conclusions and Future Work

Mathematical models of infectious disease systems are used to address specific and important problems in the dynamics, persistence and control of those systems. Here we have brought together two contrasting modelling approaches: a population based approach from mathematics and an individual based approach from computing science, and applied them to epidemiological problems.

We have examined a simple SIR system (with a closed population), comparing the use of process algebra (WSCCS) with the use of differential equations. Our main aim was to test the use of process algebras in describing infectious disease systems since they may provide an accessible method of deriving individual based models that may be analysed algebraically to yield population level results. Initial studies have shown process algebra to be an intuitive and useful way to describe disease systems. To fully investigate the questions posed above, we need to consider more complex models with decreasing levels of algebraic

tractability (from the viewpoint of traditional analyses). We are also considering the use of other process algebras for this work; in particular a continuous time model could be more realistic, and certainly be more comparable with traditional models. A suitable formalism in which to write such models would be the Performance Evaluation Process Algebra (PEPA) [8].

We have shown here that it is possible to chose rules of individual behaviour that give us the same population level behaviour as the established ODE approach. At this point we are establishing the theoretical basis for future work, which will throw into contrast the different assumptions made in each modelling exercise as more complex systems are considered. Our studies have raised theoretical questions about the best way to write the rules down. For example, the need to separate choice from communication is clear, while the use of the priority operator to force communication seems at present to be optional (depending on the form of the other rules). These questions are likely to become more important as we try to add more realism.

The next step is to add births and deaths to this simple system. This is where the interesting dynamics lie. This will require formulation of explicit rules giving density dependent transmission since the total population will no longer be constant. We also intend to carry out further analysis, for example exploring the Markov chain behaviour.

References

1. Anderson, R., May, R.: The population dynamics of microparasites and their invertebrate hosts. *Philosophical transactions of the Royal Society* **291** (1981) 451–524
2. Rand, D., Keeling, M., Wilson, H.: Invasion, stability and evolution to criticality in spatially extended, artificial host-pathogen ecology. *Proceedings of the Royal Society of London Series B* **259** (1995) 55–63
3. Sato, K., Matsuda, H., Sasaki, A.: Pathogen invasion and host extinction in lattice structured populations. *Journal of Mathematical Biology* **32** (1994) 251–268
4. Boots, M., Sasaki, A.: 'Small worlds' and the evolution of virulence: infection occurs locally and at a distance. *Proceedings of the Royal Society of London Series B* **266** (1999) 1933–1938
5. Tofts, C.: Processes with probabilities, priority and time. *Formal Aspects of Computing* **6** (1994) 536–564
6. Tofts, C.: Describing social insect behaviour using process algebra. *Transaction of the Society for Computer Simulation* (1993) 227–283
7. Sumpter, D.: From Bee to Society: an agent-based investigation of honeybee colonies. PhD thesis, UMIST (2000)
8. Hillston, J.: *A Compositional Approach to Performance Modelling*. Cambridge University Press (1996)

On Representing Biological Systems through Multiset Rewriting

S. Bistarelli^{1,2}, I. Cervesato^{3*}, G. Lenzini⁴, R. Marangoni^{5,6}, and F. Martinelli¹

¹ Istituto di Informatica e Telematica – C.N.R.

Via G. Moruzzi 1, I-56100 Pisa – Italy

`{fabio.martinelli,stefano.bistarelli}@iit.cnr.it`

² Dipartimento di Scienze, Università di Pescara,

Viale Pindaro 87, I-65127 Pescara – Italy

`bista@sci.unich.it`

³ Advanced Engineering and Sciences Division, ITT Industries, Inc.

Alexandria, VA 22303 – USA

`iliano@itd.nrl.navy.mil`

⁴ Istituto di Scienza e Tecnologie Informatiche – C.N.R.

Via G. Moruzzi 1, I-56100 Pisa – Italy

`lenzini@iei.pi.cnr.it`

⁵ Istituto di Biofisica – C.N.R.

Via G. Moruzzi 1, I-56100 Pisa – Italy

`roberto.marangoni@ib.pi.cnr.it`

⁶ Dipartimento di Informatica, Università di Pisa

Via F. Buonarroti 2, 56127 Pisa – Italy

Abstract. We model qualitative and quantitative aspects of metabolic pathways by using a stochastic version of Multiset Rewriting (SMSR). They offer a natural way of describing both the static and the dynamic aspects of metabolic pathways. We argue that, due to its simple conceptual model, SMSR may be conveniently used as an intermediate language where many higher level specification languages may be compiled (e.g., as in the security protocol example). As a first step, we show also how SMSR may be used to simulate Stochastic Petri Nets for describing metabolic pathways.

1 Introduction

In the post-genomic era, the most prominent biological problems are detecting, describing and analyzing the informational flows that make a set of molecules a living organism [1]. Genomic and proteomic techniques, in fact, are producing the largest set of biological data available ever, but the problem of detecting and describing how these entities (genes and proteins) interact with each other in the complex molecular machinery of the cell has just begun being addressed. It is

* Cervesato was partially supported by NRL under contract N00173-00-C-2086. This work was completed while this author was visiting Princeton University.

necessary to find easy, comprehensive, and biological-friendly *models* to describe molecules and their interactions.

Metabolism can be defined as the sum of all the enzyme-catalyzed reactions occurring in a cell. There are relatively few metabolic pathways, but each of these can be broken down into many individual, enzyme-specific, catalyzed steps. Metabolism is a highly integrated process. Individual metabolic pathways are linked into complex networks through common, shared substrates. A series of nested and cascaded feedback loops are employed to allow flexibility and adaptation to changing environmental conditions and demands. Negative feedback (usually by end-product inhibition) prevents the over-accumulation of intermediate metabolites and it contributes to maintaining homeostasis.

Understanding the mechanisms involved in metabolic regulation has important implications in both biotechnology and in medicine. For example, it is estimated that at least a third of all serious health problems such as coronary heart disease, diabetes and strokes are caused by metabolic disorders. Due to the integrated nature of metabolism, it is often difficult to predict how changing the activity of a single enzyme will affect the entire reaction pathway. Mathematical kinetic models have been applied to help elucidate the behavior of biochemical networks.

It is common opinion [1] that an ideal model for biological enquiring has to satisfy three requirements:

- It must be suitable for describing metabolic networks, in order to create metabolic databases allowing the user to search for and compare biochemical pathways in living organisms (like the genomic and proteomic database are already doing).
- It must be implementable into a simulation machine, in order to realize dynamic models of metabolic pathway that allow studying possible critical situation and steady states, and generally predicting that certain conditions will happen.
- It must be possible to run dynamic simulations in which to evaluate how external agents interfere with molecules and processes, in order to infer the consequences on the metabolic network stability. This kind of applications is a useful in silico test of possible side effects of a drug.

For these reasons, proper theories and instruments of the Formal Methods research community may help in defining formal models and tools (*e.g.*, see [2]), since they have been used so far to represent different kinds of relationships and dynamic interactions among objects and processes in distributed systems. In this paper, we use Multiset Rewriting (MSR) [3,4], a logic-based formalisms based on rewriting systems. MSR offers both a formal language for a precise description of molecular interaction maps, and an execution model allowing simulation of the dynamics of molecular networks with the theoretical possibility of predicting optimal values for certain parameters used in the system description. Basic mechanisms in MSR include: (a) a multiset of items, used to describe a system state, which can represent objects or resources or generic entities; (b) a set of rewriting rules which act on a state by consuming and producing items. It is

our opinion that those simple and abstract mechanisms are expressive enough in describing a large class interactions happening in molecular systems.

The rest of the paper is organized as follows. Section 2 and 3 recall respectively the multiset rewriting framework and its stochastic extension. The main result of the paper is described in Section 4 where biochemical systems are modeled as multiset rewriting rules. Section 5 gives a complete real example showing the applicability of the framework. Finally, Section 6 show a theoretical results that gives the possibility to transform biological systems represented as petri nets in our MSR model. Section 7 summarize the results and highlight some future related research topics.

2 Multiset Rewriting

The formal language of MultiSet Rewriting, MSR [4,3], is given by the following grammar, defining multisets, multiset rewriting rules and rule sets:

$$\begin{array}{ll}
 \textit{Multisets} & \tilde{a}, \tilde{b}, \tilde{c}, \tilde{g} ::= \cdot \mid a, \tilde{a} \\
 \textit{Multiset rewrite rules} & r ::= \tilde{a} \rightarrow \tilde{b} \\
 \textit{Rule sets} & \tilde{r} ::= \cdot \mid r, \tilde{r}
 \end{array}$$

The elements of a multiset, denoted a above, are *facts* $p(\mathbf{t})$ where p is a predicate symbol and the terms $\mathbf{t} = (t_1, \dots, t_n)$ are built from a set of symbols Σ and variables $x, y, z, \dots$. Numerous examples will be given in the sequel. The elements in a multiset $\tilde{a}$ shall be considered unordered, but may contain replicated elements. For convenience, “.” will be kept implicit when $\tilde{a}$ has at least one element. Similar conventions apply to rule sets.

In a rule $r = \tilde{a} \rightarrow \tilde{b}$, the multisets $\tilde{a}$ and $\tilde{b}$ are called the *antecedent* and the *consequent*, respectively. We will sometimes emphasize that the above rule mentions variables $\mathbf{x} = (x_1, \dots, x_n)$ by writing it $r(\mathbf{x}) = \tilde{a}(\mathbf{x}) \rightarrow \tilde{b}(\mathbf{x})$. Then, we denote the rule obtained by substituting the variables $\mathbf{x}$ with terms $\mathbf{t} = (t_1, \dots, t_n)$ as $r(\mathbf{t}) = \tilde{a}(\mathbf{t}) \rightarrow \tilde{b}(\mathbf{t})$.

An MSR specification describes the situation a system is in at a certain instant as a multiset $\tilde{a}$ without any variable. This is called a *state* and written s possibly subscripted. The transformations that describe the legal evolution of the system are given as a set of rules $\tilde{r}$. We represent the fact that the system evolves from state s to state s' by using one rule r in $\tilde{r}$ as the judgment

$$\textit{Single rule application} \quad \tilde{r} : s \longrightarrow s'$$

Operationally, this step is described by the following inference rule:

$$\frac{}{(\tilde{r}, \underbrace{\tilde{a}(\mathbf{x}) \rightarrow \tilde{b}(\mathbf{x})}_r) : \underbrace{\tilde{c}, \tilde{a}(\mathbf{t})}_s \longrightarrow \underbrace{\tilde{c}, \tilde{b}(\mathbf{t})}_{s'}}$$

In order for r to be *applicable* in s , this state must contain an instance $\tilde{a}(\mathbf{t})$ of r 's antecedent $\tilde{a}(\mathbf{x})$, and possibly some other facts $\tilde{c}$. If r is applicable, s' is obtained

from s by removing $\tilde{a}(\mathbf{t})$ and replacing it with the corresponding instance of the consequent, $\tilde{b}(\mathbf{t})$. Basic execution steps can be chained. The iterated judgment is written $\tilde{r} : s \longrightarrow^* s'$.

3 Stochastic MSR

Stochastic MSR (SMSR) is an extension of MSR aimed at studying reductions quantitatively. The duration of each reduction is exponentially distributed. The rate of that reduction is given as a result of applying a weight function w to the current state s . A stochastic MSR rule with weight w is denoted $\tilde{a}(\mathbf{x}) \rightarrow_w \tilde{b}(\mathbf{x})$. The notion of rule application is modified as follows:

$$\overline{(\underbrace{\tilde{r}, \tilde{a}(\mathbf{x}) \rightarrow_w \tilde{b}(\mathbf{x})}_r : \underbrace{\tilde{c}, \tilde{a}(\mathbf{t})}_s \longrightarrow_{w(s)} \underbrace{\tilde{c}, \tilde{b}(\mathbf{t})}_{s'})}$$

Note that the rewriting rule naturally determines a labeled transition systems *LTS* whose states are the multisets and whose transitions are the reduction rule instances together with their rates. A so-called *race condition* determines the dynamic behavior of the system, i.e. when more of different rules are enabled, only the fastest succeed in being fired. It is worth to note that the continuous character of the exponential distribution assures that the probability of two rules firing at the same time is zero. The race condition has the effect of replacing the *possibilistic* structure of the underlying *LTS* into a *probabilistic* where the probability of each transition is proportional to its rate. This means that each rule will fire not only when the head of a rule unify with (part of) a the current states, but also depending on other conditions implemented with the function $w(s)$. Specific characteristics of the system can be analyzed instantiating a specific w function (representing the race condition).

The analogy between biochemical reactions and SMSR is given in Table 1.

Table 1. Analogy between Biochemical Notions and SMSR.

Predicate name	molecular species
Predicate	molecule
Rewriting	reaction
To be enabled	for a reaction to be possible
To fire	for a reaction to occur
Weight	reaction rate

4 Modeling Biochemical Systems with SMSR

Biochemical Reactions. Biochemical reactions are usually represented by the following notation:



where A, B, C and D are molecules, $a, b, c, d, \dots$ are their stoichiometric coefficients (which are constants), and k, k_{-1} are the kinetic constants. The previous formula may be considered as a declaration of the different proportions of *reactants* and *products*, namely the objects at the left, and respectively at the right of a rule $\multimap^k$. This proportion is given by the following formula:

$$K = \frac{k}{k_{-1}} = \frac{[C]^c [D]^d}{[A]^a [B]^b}$$

which expresses the equilibrium constant, and $[A], [B], \dots$ are the *concentrations* (*i.e.*, , moli over volume unit) of the respective molecules. The reaction rate (*i.e.*, the number of moli produces per time unit) depends usually on the kinetic constant and on the concentration of the reactants. In some situations the reaction rate may be influenced by other entities that slow its rate.

We can note that aA may be simply considered as:

$$\overbrace{A + \dots + A}^a$$

and so the whole equation 1 may be considered a shortcut of for

$$\overbrace{A + \dots + A}^a + \overbrace{B + \dots + B}^b \xrightleftharpoons[k_{-1}]{k} \overbrace{C + \dots + C}^c + \overbrace{D + \dots + D}^d$$

This intuition give a first idea about how a reaction can be modeled in SMSR. We encode each reactions in the two directions as two separate SMSR rules. For example equation 1 may be encoded in the two following SMSR rules:

$$\overbrace{A, \dots, A}^a, \overbrace{B, \dots, B}^b \rightarrow_{w_k} \overbrace{C, \dots, C}^c, \overbrace{D, \dots, D}^d$$

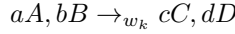
and

$$\overbrace{C, \dots, C}^c, \overbrace{D, \dots, D}^d \rightarrow_{w_{k_{-1}}} \overbrace{A, \dots, A}^a, \overbrace{B, \dots, B}^b$$

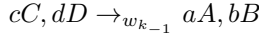
where w_k and $w_{k_{-1}}$ are the functions (which here depend on the kinetic values k and k_{-1} , on the stoichiometric coefficients a, b, c and d , and on the overall number of predicates A, B, C and D currently in the state defining the application rate of the rule. With abuse of notation, if we consider that

$$aA \equiv \overbrace{A, \dots, A}^a$$

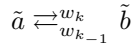
then the equation (1) may be very naturally expressed by the following two rewriting rules:



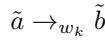
and



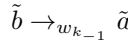
where w_k, w_{k-1} are the functions that will define the actual reaction rate of the reduction depending on several factors, as the kinetic constants, and the concentrations/quantities of reactants. (Indeed, as stated in [5,2], under certain assumptions concentration and quantities may be exchanged.) If we use the notation



to abridge



and



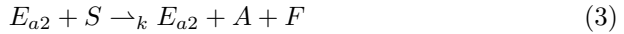
then the equation 1 may be represented in the SMSR framework as:



that nicely resembles the traditional notation.

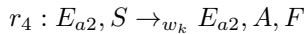
Enzymatic reactions are described similarly. The enzyme is considered as a persistent predicate that is not consumed by the reaction:

For example in the urea cycle (see Figure 1), the stoichiometric reaction (3)



where the production of a certain amount of *Arginine* A and *Fumarate* F , from *Arginsuccinate* S and catalyzed by the enzyme *Arginosuccinase* E_{a2} , is controlled by the concentration of the *Arginine* in the environment. Precisely if the *Arginine* concentration strongly increases the probability of this reaction decreases.

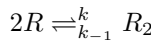
Reaction 3 may be expressed in SMSR with the following rule:



where the reaction rate $w_k = f(k, [A])$ is an inverse function depending on the kinetic constant k and on the number of predicates A (thus the quantity of arginine) in the actual state.

4.1 A Small Example

Consider the example in [6] about the dimerization reaction of a molecule R . Assume that the biochemical rule is:



also written as

$$R + R \rightleftharpoons_{k_{-1}}^k R_2$$

With our formalism, it has been split in the two rules:

$$R, R \rightarrow^{w_k} R_2$$

and

$$R_2 \rightarrow^{w_{k-1}} R, R$$

or equivalently

$$R, R \xrightleftharpoons[w_{k-1}]{w_k} R_2$$

During the construction of the rule we need to build the rate function of the reduction. This rate is usually function of concentrations or quantities. In a monomolecular reaction, the weight function is given $w_{k-1} = c_{k-1} \cdot |R_2|$, i.e. the product among the quantity of molecules and a constant depending on the kinetic one k . In higher order molecular reaction, the weight function is $w_k = c_k \cdot |R| \cdot (|R| - 1)$, where c_k depends on the kinetic constant k .

5 The Urea Cycle: A Complete Modeling Example

This section shows how to express in SMSR the reactions in the urea cycle. In particular we refer to the pathway in Figure 1, representing the main interaction occurring the human urea cycle.

In the urea cycle a sequence of chemical reactions, occurring primarily in the liver, the ammonia is converted to urea in mammalian tissue. The urea, far less toxic than ammonia, is subsequently excreted in the urine of most mammals. Also known as the ornithine-citrulline-arginine-urea cycle, this cycle also serves as a major source of the amino-acid arginine.

Assume the following abbreviations:

Table 2. Abbreviations

Enzymes		Molecules			
<i>Arginase</i>	E_{a1}	<i>Arginine</i>	A	<i>Arginosuccinate</i>	S
<i>Arginosuccinase</i>	E_{a2}	<i>Aspartate</i>	P	<i>Carbamoylphosphaphate</i>	C_p
<i>Arginosuccinase Synthase</i>	E_{a3}	<i>Citrulline</i>	C_i	<i>Fumarate</i>	F
<i>Ornithine Transcarbamoylase</i>	E_o	<i>Ornithine</i>	O	<i>Urea</i>	U
		<i>Water</i>	H_2O		

the reactions in the urea cycle are described by the stoichiometric equations in Table 3, where both a continuous production of H_2O and C_p , and a destruction of U are assumed in the environment. Here we assume these environmental condition to be guaranteed by external reaction (that we express as [5-7] in Table 4). The kinetic constants k_i (here left unspecified) define the rate of the relative reactions. Usually these rates are calculated experimentally by biologist and their exact values are available in the literature or retrieved from one of the public database on the web (e.g., from KEGG pathways database).

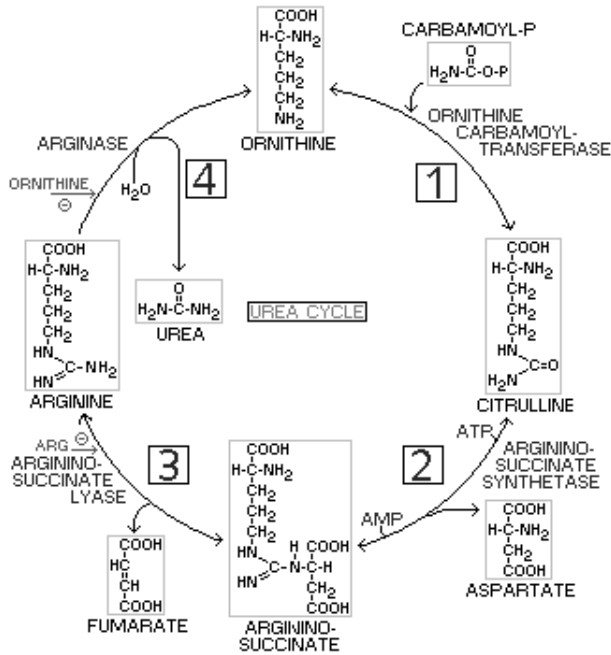


Fig. 1. The Urea Cycle

In the urea cycle two fundamentals feed-back regulations contribute to the right production of *Arginine* and *Ornithine*. Informally they can be described as in the following:

1. if *Arginine* concentration strongly increases, the probability of reaction [3] decreases;
2. if *Ornithine* concentration strongly increases, the probability of reaction [4] decreases;

These mean that the *Arginine* and *Ornithine* production directly controls their own rate of production. For example an excessive production of *Arginine* causes negative feedback on the reaction producing the *Arginine* itself, decreasing the probability of that reaction to happen.

Using SMSR, the stoichiometric reactions in Table 3 may be quite literally expressed as rewriting rules and the predicate symbols used to represent molecular entities are exactly the one used in the stoichiometric equations. In Table 4 we report the SMSR rewriting rules which model the urea cycle.

In the rewriting rules in Table 4 it worth stressing that the catalyzing enzymes are expressed as persistent predicates, i.e. they are not consumed in practice (clearly we can model also other situations). The stochastic parameters w_{k_i} are indeed function of the kinetic constant k_i , or the stoichiometric coefficients c (here all equal to 1), of the relative stoichiometric equation.

Table 3. Stoichiometric equations for the urea cycle.

- [1] $E_o + O + C_p \xrightleftharpoons[k_{-1}]{k_1} C_t + E_o$
- [2] $E_{a3} + C_t + P + ATP \xrightleftharpoons[k_{-2}]{k_2} S + AMP + E_{a3}$
- [3] $E_{a2} + S \xrightleftharpoons[k_{-3}]{k_3} A + F + E_{a2}$
- [4] $E_{a1} + A + H_2O \xrightarrow{k_4} U + O + E_{a1}$

Table 4. SMSR rules for the urea cycle.

- [1] $E_o, O, C_p \xrightleftharpoons[w_{k_{-1}}]{w_{k_1}} C_t, E_o$
- [2] $E_{a3}, C_t, P, ATP \xrightleftharpoons[w_{k_{-1}}]{w_{k_2}} S, E_{a3}$
- [3] $E_{a2}, S \xrightleftharpoons[w_{k_{-3}}]{w_{k_3}} A, F, E_{a2}$
- [4] $E_{a1}A, H_2O \xrightarrow{w_{k_4}} U, O, E_{a1}$
- [5] $\cdot \rightarrow C_p$
- [6] $\cdot \rightarrow H_2O$
- [7] $U \rightarrow \cdot$

Generally speaking we can assume that, for a stoichiometric equation $[i]$, $w_{k_i} = f_i(k_i, c_i, s)$, where f_i is a monotonic increasing function coming from biological enquiring. For example the regulation feedback of the urea cycle forces the following definitions:

$$\begin{aligned}
 w_{k_3} &= f_3(k_3, |A|^{-1}) \\
 w_{k_{-3}} &= f_{-3}(k_{-3}, |A|) \\
 w_{k_4} &= f_4(k_4, |O|^{-1}) \\
 w_{k_{-4}} &= f_{-4}(k_{-4}, |O|)
 \end{aligned}$$

where we want to express that the value of w_{k_3} (resp. w_{k_4}) inversely depend on the number of predicates A (resp. O) in the global state¹. Similarly $w_{k_{-3}}$ (resp. $w_{k_{-4}}$) directly depends on the number of predicate A (resp. O) in the global state.

6 Simulating Stochastic Petri Nets for Metabolic Pathways Modeling with SMSR

In this section we show how SMSR can simulate the analysis performed on metabolic pathways done using Stochastic Petri Nets (SPN's) [6]. We argue that this is not an isolate case and many other modeling approaches based on formalisms for distributed and concurrent systems may be encoded in our framework.

Petri Nets (PNs) are a family of distributed calculi, based on the notions of *places*, *tokens*, *markings* and *transitions*. The idea is that places stores resources,

¹ Here $|A|$ returns the number of occurrences of the predicates A in s

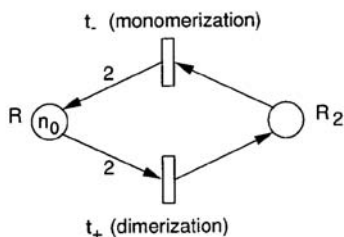


Fig. 2. SPN representation of the dimerization of molecule R

i.e. tokens. A marking is an instantaneous picture of the tokens present in the places. A transition is a relation among set of places to set of places and describes how markings change during the computation. A link from a place to a transition is equipped with a number which expresses how many token are necessary from that place to enable the transition. A link from a transition to a place is equipped with a number that expresses how many token are produced when the transition is fired. Transitions in stochastic Petri Nets are not instantaneous (as in many other PNs), have a duration determined by a given probability distribution, usually exponential. This duration is inserted in the transition.

As an example [6], consider the dimerization reaction of the molecule R as represented in Section 4.1, and whose corresponding SPN is represented in figure 2.

In [6], it has been advocated that Stochastic Petri Nets may be very useful when considering dozen (hundreds) of different species but a small absolute number of molecule. The analogy among metabolic pathways used in [6] is given in Table 5.

The computation of an SPN's is a graph that shows how markings evolve in time. There is a weighted link from a marking to another whenever there is a transition enabled in the previous marking that, due to its. Note that, as usual, two transitions are not assumed to fire at the same time. The weight of the transition, which clearly, denotes the rate of a reaction may depend on the global marking.

We can propose an analogy among SPN's and SMSR as follows: Places correspond to Predicate names; a token in a place is a predicate of a certain kind in the multiset and thus a marking is essentially a multiset of atomic predicates. Transitions are encoded as rewriting rules: A transition that consumes n token from a certain place is encoded through a rule which requires n instances of the predicate corresponding to the place in the left hand side; a transition that produces m token in a place corresponds to a rule which requires m predicates on the right end side. The weight function of the rewriting rule is the same of the transition, provided the analogy among the marking and the multiset.

Thus, the *LTS* produced by the SMSR from an initial multiset precisely corresponds to the computation graph of the initial marking.

Table 5. Analogy between biochemical notions and SPN's.

Place	molecular species
Token	molecule
Marking	quantities for each molecular species
Transition	reaction
Transition enabled	for a reaction to be possible
Transition fired	for a reaction to occur

7 Conclusions, Future, and Related Work

The MSR formalism has been used to study several forms of concurrent distributed computation [4,7]. We advocate a simple stochastic variant of MSR, named SMSR, as a natural framework to model biochemical reactions. It has a clear and rigorous formal semantics. Moreover, its rules are readily understandable and seem quite close to textual descriptions of chemical reactions, at a functional level. From the specification and analysis point of view, we started to explore a line of research already successfully followed for security protocols analysis. Indeed, many analysis tools compile high level specifications into simpler MSR specifications [8]. Due to the simplicity and uniformity of the MSR rewrite mechanism, many researchers feel it is easier to develop analysis algorithms for MSR. In this paper, we show that a well known model of analysis may be encoded into the SMSR, i.e. SPN's. We argue that it will be possible to faithfully encode also other specification languages as the stochastic process algebra [2] into stochastic MSR (similarly to the encoding we proposed in [7] for security analysis). An advantage is that the intermediate language is itself a significant one for biologists.

Future work. We have two main goals:

- We plan to produce an optimized simulation environment based on the input syntax of *SMSR* (possibly refined with built-in predicates or constraints) and map into other specification formalisms;
- We plan also to adapt the rich theory already developed for *MSR* for a suitable definition within biological systems of very useful formal notions like composition, abstraction, equivalence, congruences and so on.

Related work. Other notable approaches using forms of (multiset)-rewriting are presented in [9,10]. The first approach exploits the modeling system Maude based on algebraic notions and rewrite theory. The structure and hierarchy of biological elements are represented through terms from a rich algebra. The emphasis is more on the qualitative aspects of the interactions. The latter is closer to ours since it is mainly based on simulation of biochemical reactions. The language, however, does not allow generic terms as ours. Moreover, our work instead is more focused on showing the ability of SMSR of acting as natural low level language for more complex description languages.

References

1. Pevzner, A.: Computational Molecular Biology. MIT press (2000)
2. Priami, C., Regev, A., Shapiro, E., Silverman, W.: Application of a stochastic name-passing calculus to representation and simulation of molecular processes. *Information Processing Letters* **80** (2001)
3. Cervesato, I., Durgin, N., Lincoln, P.D., Mitchell, J.C., Scedrov, A.: A Meta-Notation for Protocol Analysis. In: 12th Computer Security Foundations Workshop – CSFW-12, Mordano, Italy, IEEE Computer Society Press (1999) 55–69
4. Cervesato, I.: A Specification Language for Crypto-Protocols based on Multiset Rewriting, Dependent Types and Subsorting. In Delzanno, G., Etalle, S., Gabrielli, M., eds.: Workshop on Specification, Analysis and Validation for Emerging Technologies – SAVE’01, Paphos, Cyprus (2001) 1–22
5. Gillespie, D.: Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry* **81** (1977) 2340–2361
6. Goss, P., Peccoud, J.: Quantitative modeling of stochastic systems in molecular biology by using stochastic Petri Nets. In: Proc. National Academy of Sciences USA. Volume 95. (1998) 6750–6754
7. Bistarelli, S., Cervesato, I., Lenzini, G., Martinelli, F.: Relating process algebras and multiset rewriting (for example for security protocol analysis). Technical report, Istituto di Scienza e Tecnologie dell’Informazione (ISTI-CNR) (2002) To appear.
8. Denker, G., Millen, J.K.: CAPSL Intermediate Language. In Heintze, N., Clarke, E., eds.: Proceedings of the Workshop on Formal Methods and Security Protocols – FMSP, Trento, Italy (1999)
9. Eker, S., Knapp, M., Laderoute, K., Lincoln, P., Meseguer, J., Sonmez, K.: Pathway logic: Symbolic analysis of biological signaling. In: Proc. Pacific Symposium on Biocomputing. Volume 7. (2002) 400–412
10. R. Hofestädt, M.L.u.U.S.: Molecular information fusion for metabolic networks. In: In G.X. Xue, Y.B. Xue, Z.H. Xu, R. Holmes, G. Hammond und H.A. Lim, Herausgeber, Gene Families: Studies of DNA, RNA, Enzymes and Proteins. (2001) 221–232

A Model of Neural Inspiration for Local Accumulative Computation

José Mira¹, Miguel A. Fernández², Maria T. López², Ana E. Delgado¹,
and Antonio Fernández-Caballero²

¹ Departamento de Inteligencia Artificial
Facultad de Ciencias y E.T.S.I. Informática, UNED,
28040 - Madrid, Spain
{jmira, adelgado}@dia.uned.es

² Departamento de Informática
E.P.S.A., Universidad de Castilla-La Mancha,
02071 – Albacete, Spain
{miki, mlopez, caballer}@info-ab.uclm.es

Abstract. This paper explores the computational capacity of a novel local computational model that expands the conventional analogical and logical dynamic neural models, based on the charge and discharge of a capacity or in the use of a D flip-flop. The local memory capacity is augmented to behave as an S states automaton and some control elements are added to the memory. The analogical or digital calculus equivalent part of the balance between excitation and inhibition is also generalised to include the measure of specific spatio-temporal features over temporal expansions of the input space (dendritic field). This model is denominated as accumulative computation and is inspired in biological short-term memory mechanisms. The work describes the model's general specifications, including its architecture, the different working modes and the learning parameters. Then, some possible software and hardware implementations (using FPGAs) are proposed, and, finally, its potential usefulness in real time motion detection tasks is illustrated.

1 Introduction

The most usual analogical models in neural computation are of a static nature. Once the input values in an instant, $\bar{x}(t)$, and the values of the weights, $\bar{\omega}(t)$, are known, the output value in that instant, $\bar{y}(t) = \bar{\omega}(t) \cdot \bar{x}(t)$, is obtained. Nevertheless, one important part of the biological processes and of the proper computation are rather of a dynamic nature; that is to say, they are models dependent of time where the response, $\bar{y}(t)$, is a function of the inputs and responses in earlier instants, $\{\bar{x}(t - K_1 \cdot \Delta t), \bar{y}(t - K_2 \cdot \Delta t)\}$. In order to model these dynamic nets a set of state variables described by a first order differential equation $\tau_j \frac{dy_i(t)}{dt} = -y_i(t) + h_j$, are

introduced, such that in the stationary case variable $\bar{y}(t)$ reaches its equilibrium value (h_j) with a time constant τ_j .

When adding the effect of the inputs $\{x_i(t)\}$, the linear part of the expression of a dynamical neural model known as leaky integrator is gotten. This means that the value and the sign of the state variable depend on the excitation and inhibition in the receptive field of the calculus element:

$$\tau_j \frac{dy_j(t)}{dt} = -y_j(t) + \sum_{i \in V_j} w_{ji} \cdot x_i(t) + h_j$$

In this case the influence of the temporal component of the calculus (the analogical memory) is physically represented by means of charge and discharge processes of a capacitor [10].

On the other hand, in digital models of neural networks, local memory is introduced by means of a D flip-flop that represents the effect of the synaptic delay [11]. In this case the computational model is a modular sequential circuit (a modular automaton) in which each calculus element ("neurone") is a universal two states automaton, which may calculate any logical function of its inputs and of the proper and other neurones outputs in the previous instant,

$$y_j(t + \Delta t) = \sum_{i=0}^{2^{N+M}-1} \omega_{ij}(t) \cdot m_i(t), \text{ where } \omega_{ij}(t) \in \{0, 1\}, \text{ are the binary weights and}$$

$m_i(t)$ are the minterms, $m_i = x_1^\alpha \cdot x_2^\beta \cdots x_M^\mu \cdots y_N^\gamma$ for $i = \alpha\beta \cdots \mu \cdots \gamma$,

using Gilstrap notation: $(x^0 = \bar{x}, x^1 = x)$

This paper explores the computational capacity of a novel local computational model that expands the conventional analogical and logical dynamic neural models, based on the charge and discharge of a capacity or in the use of a D flip-flop. The local memory capacity is augmented to behave as an S states automaton and some control elements are added to that local memory. The analogical ($W^T \bar{x}(t)$) or digital ($\sum \omega_{ij}(t) \cdot m_i(t)$) calculus equivalent part of the balance between excitation and inhibition is generalised to include any pre-processing not related to learning where spatio-temporal features of the stimuli are calculated over temporal expansions of the input space. This expansion with a FIFO memory structure represents the computational features of the receptive field, which make computationally homogeneous the data fields coming from different time intervals. The part corresponding to the delay management is also generalised by substituting it by an S states automaton with a reversible counter structure (or a RAM memory), where the increment and decrement of its content is programmable. This model is denominated as accumulative computation.

The rest of the paper is organised in the following way. Section 2 describes the model's general specifications, including its architecture, the different operating modes and the learning parameters. Afterwards, in sections 3 and 4 some software and hardware (using FPGAs) implementations are proposed. Section 5 illustrates the potential usefulness of this local computational model in real time motion detection tasks.

2 The Model's Functional Architecture

Figure 1 shows the accumulative computation model's block diagram. The model works in two time scales, a macroscopic one, t , associated to the external data sequence to be processed by the net, and a microscopic one, τ , internal, associated to the set of internal processes that take place while the external data (an image, for instance) remain constant. The model contains the following elements:

1. A temporal extension of the input space (a FIFO memory) that permits to access the value of the inputs in various successive time instants.
2. A module of spatio-temporal features extraction over that input expansion. The measured feature is binarised and, from this moment on, the temporal accumulation of its persistency on that data field is calculated.
3. A module that calculates the increment or decrement value, $(\pm\delta Q)$, of the activity state of that property as a function of its value in that instant, $Q(t)$, of the accumulated value in the previous instants and of the accumulation mode selected in the control unit.
4. An accumulation module of reversible counter type or RAM memory, which stores the new persistency state of the selected feature.
5. A control module of the accumulation mode, which receives inputs from the programming and learning modules, and controls the operation of state changing of the memory from the calculus of increments or decrements, $\pm\delta Q$, on the previous value. There are three general operating modes for the model: (I) Initialisation, (II) calculation and (III) reconfiguration (learning). During the calculation mode, and in accordance with the temporal sequence of values $p(x,y;t)$ measured on the input data $I(x,y;t)$, one of the following processes is activated: (1) Gradual charge, (2) abrupt charge, (3) gradual discharge, (4) abrupt discharge or (5) stand-by. The parameter values that specify the charge and discharge processes (Q_{max} , Q_{min} , $+\delta Q$, $-\delta Q$) are introduced into the model during the initialisation phase and are modified during the learning phase.
6. A module of supervised learning, which enables to adjust the value of charge and discharge parameters to the shape, size and velocity features of the objects of interest that appear in the image sequences.
7. A programming module used to configure the control mode and to specify the temporal expansion of the input space and the shape of the receptive field. This way it is possible to specify the spatio-temporal property that we want to highlight alter accumulating its persistency.
8. A temporisation module consisting of a master clock, which generates the pulse train that controls the local time ("microscopic") used to calculate the value and sign of the accumulation state change and the transition to the new charge state, as well as the production of the response of the unit that passes to a FIFO for its distribution to the neighbouring modules. While the internal calculus is performed, data of the input space remain constant, controlled by the "macroscopic" clock resulting from having divided by n the frequency of the master clock.

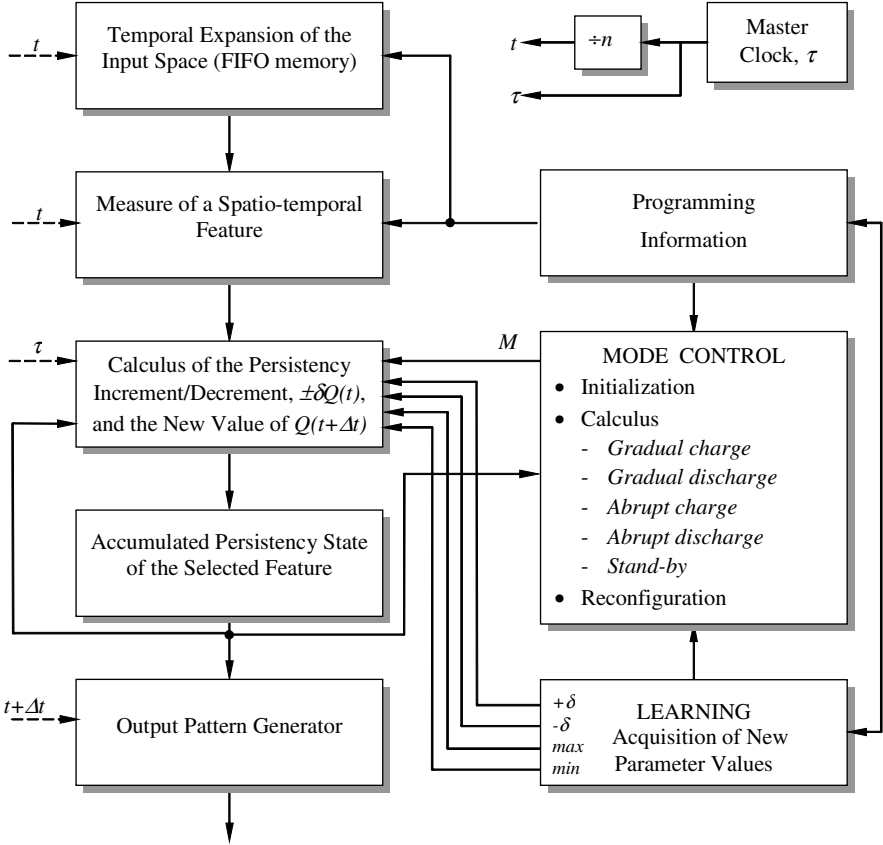


Fig. 1. Accumulative computation architecture

3 Software Simulation

In figure 2 the accumulative computation model's behaviour is shown in one-dimensional and very easy situations. Let us suppose that values of $I(x,y;t)$ correspond to an indefinite sequence of images where several objects are moving. Let us also suppose that the measured property, $p(x,y;t)$ is simply the result of the binary threshold of image $I(x,y;t)$. Then, the control mode compares values of $p(x,y;t)$ in two successive instants, interpreting that $p(x,y;t)=1$ means that there is a moving object over pixel (x,y) at t and that $p(x,y;t)=0$ means there is no mobile. Thus, changes $p(t-\Delta t)=0 \Rightarrow p(t)=1$ mean that a moving object has entered that unit's receptive field. If $p(t-\Delta t)=1$ and $p(t)=0$, a mobile has quitted the receptive field (RF); if both are zero, there is no mobile over the RF, and, finally, if both are one, there is a moving object crossing over the RF. For this property, the evolution of charge and discharge of its persistency is shown in figure 2 for some modalities of use selected.

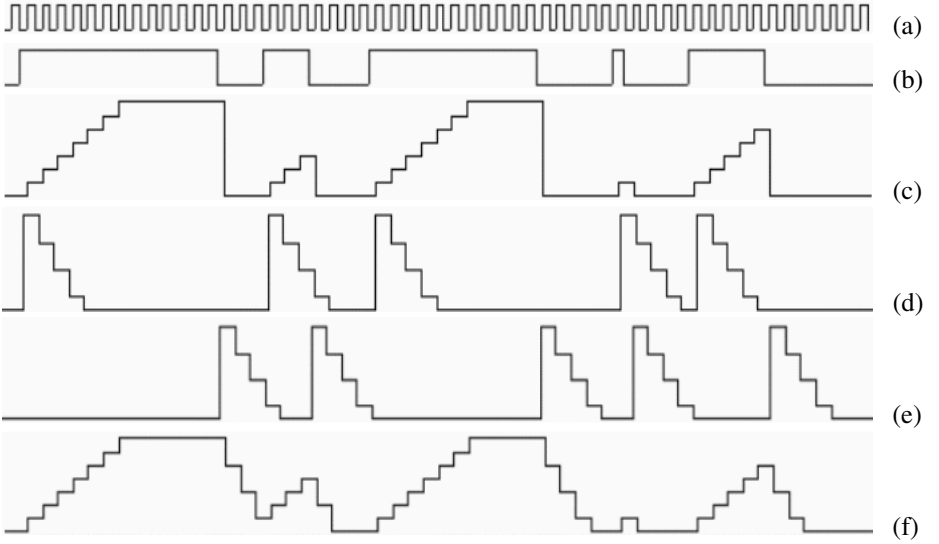


Fig. 2. Illustration of the accumulative computation model used for the easy case of binary threshold of an image. (a) Macroscopic clock t . (b) $p(t)$. (c) $Q(t)$ in LSR modality. (d) $Q(t)$ in input modality. (e) $Q(t)$ in output modality. (f) $Q(t)$ in charge/discharge modality

Figure 2c shows the behaviour of the accumulative computation model in a modality called LSR (length speed relation) [1]. This modality has been studied and used for the classification of moving objects from this relation [2]-[4].

```

if p(t) == 1
then
begin
 $Q(t) = Q(t - \Delta t) + \delta Q;$ 
if  $Q(t) > Q_{\max}$  then  $Q(t) = Q_{\max};$ 
end
else  $Q(t) = Q_{\min};$ 

```

Figures 2d and 2e show the operation of the proposed model in input and output modalities, respectively. Both options enable to perform a later calculus of characteristic motion parameters, such as velocity and acceleration [5]. The first one of these modalities offers information at the tail of the moving objects, whereas the second one does it at the front of motion. For the output modality we have:

```

if ((p(t - Δt) == 0) && (p(t) == 1))
then  $Q(t) = Q_{\max}$ 
else
begin
 $Q(t) = Q(t - \Delta t) - \delta Q;$ 
if  $Q(t) < Q_{\min}$  then  $Q(t) = Q_{\min};$ 
end;

```

In input modality we have:

```

if ((p(t-Δt) == 1) && (p(t) == 0))
  then Q(t) = Qmax
  else
    begin
      Q(t) = Q(t-Δt) - δQ;
      if Q(t) < Qmin then Q(t) = Qmin;
    end;

```

Finally, the more general charge/discharge modality is shown (figure 2f). This one has already been successfully used in some previous papers of the authors of this work [6]-[9]. These papers are about moving objects detection, classification and tracking in indefinite image sequences.

```

if p(t) == 1
  then
    begin
      Q(t) = Q(t-Δt) + δQ;
      if Q(t) > Qmax then Q(t) = Qmax;
    end
  else
    begin
      Q(t) = Q(t-Δt) - δQ;
      if Q(t) < Qmin then Q(t) = Qmin;
    end;

```

4 Hardware Implementation

The very nature of the intended calculus and the need for reconfiguration demanded by the accumulative computation model advises its implementation by means of a programmable sequential logic (e.g. field programmable gate arrays, FPGAs). These circuits contain a high number of reconfigurable identical logical modules (configurable logical blocks, CLBs) at the modules internal structure level as well as at interconnection level, and, in both cases, by simple modification of the content of a set of RAM memory cells. As an example, figure 3 shows the result of the accumulative computation model's hardware implementation on a Xilinx 4000E chip, concretely the X4003E.

Figure 3a shows the result of programming in VHDL language, and synthesising and implementing for an FPGA X4003E. In this implementation the number of different charge values $Q(t)$ has been restricted to 256. Notice that only the four modalities previously described have been implemented. Figure 3b shows in detail one of the CLBs that form the chip. The design statistics offered by the Xilinx implementation tool show a total equivalent gate count for design of 588, whereas the performance of the chip shows a minimum period value of 32.712 ns.

In this paper its usefulness in moving objects detection, classification and tracking tasks in an indefinite image sequence is illustrated. A distinctive characteristic of the approach given by this model is the possibility to compute in real time, as a consequence of the ergonomic character of the process. Figure 4 shows some of the model's capabilities introduced so far.

Indeed, figure 4 shows the model output for three significant examples. All three examples are the result of applying our model to a same image sequence, namely TwoWalkNew, downloaded from University of Maryland Institute for Advanced Computer Studies, copyright © 1998 University of Maryland, College Park. The pure output is shown on the first column for each example. The second column shows the result superimposed on the input image. The results of silhouette detection are drawn in figure 4a, motion detection in figure 4b, and direction detection in figure 4c. In this last example notice that motion direction is shown by means of the intensity of the colour. Direction has to be interpreted going from clearer to darker grey colour.

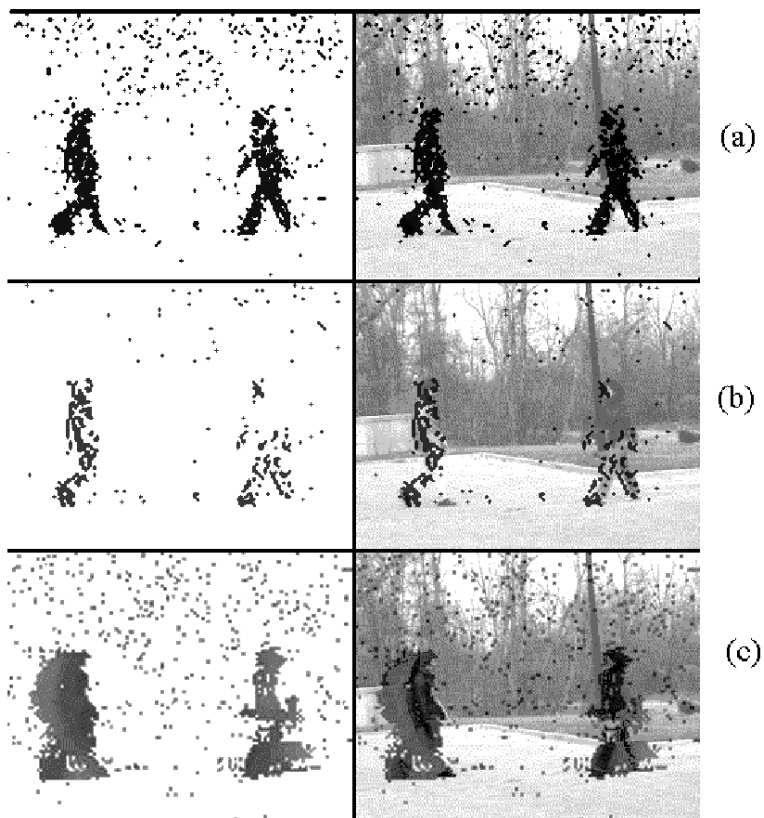


Fig. 4. Some capabilities of the accumulative computation model. (a) Silhouette detection. (b) Motion detection. (c) Direction detection

6 Conclusions

A calculus model that is modular, dynamic, of fine grain, partially self-programmable by supervised learning, and able to be integrated in a parallel architecture, has been introduced in this paper. It might be called “neuronal”, but it seems to us more adequate to consider it as a model of local calculation inspired in biological memory mechanisms. We have increased the capacity for local calculus, the memory, the features extraction as a pre-processing and the generation of output patterns. Thus, the model converts into a real time processing architecture of spatio-temporal information, based on the controlled management of a local memory.

Lastly, this paper has shown the usefulness of the accumulative computation model in artificial vision. Its use in tasks such as velocity and acceleration obtaining, moving objects detection, silhouettes detection and selective visual attention generates efficient and robust systems with competitive performances compared to any model used nowadays.

References

1. Fernández, M.A., Fernández-Caballero, A., López, M.T., Mira, J.: Length-Speed Ratio (LSR) as a characteristic for moving elements real-time classification. *Real-Time Imaging* 9:1 (2003) 49–59
2. Fernández, M.A., Mira, J.: Permanence memory: A system for real time motion analysis in image sequences. *IAPR Workshop on Machine Vision Applications, MVA'92* (1992) 249–252
3. Fernández, M.A., Mira, J., López, M.T., Alvarez, J.R., Manjarrés, A., Barro, S.: Local accumulation of persistent activity at synaptic level: Application to motion analysis. In: Mira, J., Sandoval, F. (eds.): *From Natural to Artificial Neural Computation, IWANN'95, LNCS 930*. Springer-Verlag (1995) 137–143
4. Fernández, M.A., Fernández-Caballero, A., Moreno, J., Sebastián, G.: Object classification on a conveying belt. *Proceedings of the Third International ICSC Symposium on Soft Computing, SOCO'99* (1999)
5. Fernández, M.A.: Una arquitectura neuronal para la detección de blancos móviles. Unpublished Ph.D. dissertation (1995)
6. Fernández-Caballero, A., Mira, J., Fernández, M.A., López, M.T.: Segmentation from motion of non-rigid objects by neuronal lateral interaction. *Pattern Recognition Letters* 22:14 (2001) 1517–1524
7. Fernández-Caballero, A., Mira, J., Delgado, A.E., Fernández, M.A.: Lateral interaction in accumulative computation: A model for motion detection. *Neurocomputing* 50C (2003) 341–364
8. Fernández-Caballero, A., Fernández, M.A., Mira, J., Delgado, A.E.: Spatio-temporal shape building from image sequences using lateral interaction in accumulative computation. *Pattern Recognition*, in press (2003)
9. Fernández-Caballero, A., Mira, J., Fernández, M.A., Delgado, A.E.: On motion detection through a multi-layer neural network architecture. *Neural Networks*, accepted (2003)
10. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall (1999)
11. Moreno-Díaz, R.: Realizability of a neural network capable of all possible modes of oscillation. In: Caianiello, E. (ed.): *Neural Network*. Springer-Verlag (1968) 70–78

Emergent Reasoning from Coordination of Perception and Action: An Example Taken from Robotics

Darío Maravall and Javier de Lope

Department of Artificial Intelligence
Faculty of Computer Science
Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Madrid, Spain
`{dmaravall,jdlope}@dia.fi.upm.es`

Abstract. The paper presents a manipulator arm that is able to acquire primitive reasoning abilities from the pure coordination of perception and action. First, the problem of dynamic collision avoidance is considered, as a test-bed for autonomous coordination of perception and action. The paper introduces a biomimetic approach that departs from the conventional, analytical approach, as it does not employ formal descriptions of the locations and shape of the obstacles, nor does it solve the kinematic equations of the robotic arm. Instead, the method follows the perception-reason-action cycle and is based on a reinforcement learning process guided by perceptual feedback. From this perspective, obstacle avoidance is modeled as a multi-objective optimization process. The paper also investigates the possibilities for the robot to acquire a very simple reasoning ability by means of *if-then-else* rules, that transcend its previous reactive behaviors based on pure interaction between perception and action.

1 Introduction

The last decade has witnessed the emergence of a novel paradigm for designing and building autonomous robots, usually known as the reactive paradigm [1], which differs from the traditional so-called deliberative paradigm [2] as regards the role played by the formal representation of the robot environment. The competition between the two paradigms partly coincides, within the robotics field, with the contemporaneous debate between the symbolic approach to artificial intelligence [3] and several alternative approaches, like connectionism [4] and the so-called dynamic perspective of cognitive science [5]. In fact, both debates have their roots in the old epistemological debate between rationalism and empiricism, which is also a central dispute in the very general field of what can be called science of intelligence, as it divides several schools of thought on how to understand intelligence and, in particular, on the different roles played in intelligence by apriorism, adaptation and learning. We are referring to knowledge representation and acquisition, as well as to knowledge use or, more specifically, reasoning,

which can be considered one of the three basic components of intelligence along with perception and action.

In this paper, we present a robotic mechanism that is able to acquire very primitive reasoning abilities from the pure coordination of perception and action. The robot structure is materialized by means of a bio-inspired architecture, in which perceptions from the environment and robot actions manage to autonomously coordinate themselves, chiefly through environment responses. More specifically, we have considered the hard problem of dynamic collision avoidance—i.e., the type of task that for a conventional robot requires complete, a priori knowledge of the environment—as the test-bed for our robot. We demonstrate in practice two interesting features of our approach as regards the conventional, model-based approach: (1) the robot actions are performed without solving the kinematics equations and (2) there is no need for the robot to have a formal representation of the environment in order to negotiate complex environments and avoid obstacles.

To our knowledge, the cerebellar model articulation controller (CMAC), proposed by Albus in the seventies, is the first example of a robot controlled without explicitly employing its kinematic expressions. It is based on learning the mapping functions, given as look-up tables, between the working space—i.e., Cartesian coordinates—and the configuration space—i.e., joints coordinates—by means of visual feedback. This line of research was pursued mainly in the field of cognitive science and produced interesting simulated robots and a few physical prototypes. Its reliance on look-up tables, mapping the sensory information to the control actions, makes this paradigm very prone to combinatorial explosion, unless a local search in the mapping functions is applied. The method proposed in this paper, however, differs significantly, as it is based on temporal and spatial utility functions, so that the robot performs the mapping between sensory information and actions by optimizing what we call utility functions, which dramatically reduces the search space.

The above-mentioned features can be considered as merely sophisticated types of coordination of perception and action, not as true reasoning. Thus, we also investigate the possibilities for the robot to acquire a simple reasoning ability by injecting an additional structure into the robot structure to embody simple, although powerful, *if-then-else* rules. This enables the robot to transcend its previous reactive behaviors based on pure interaction between perception and action, which can be considered as a primitive reasoning activity of a sort. Learning new strategies for obstacle avoidance illustrate the potentiality added to the robot by such reasoning ability.

2 The Robotic Manipulator Arm

We are going to illustrate our bio-inspired approach with a R-R planar manipulator arm, as depicted in Fig. 1.

Any possible movement of the robot can be performed by generating appropriate trajectories of its two generalized coordinates θ_1 and θ_2 . Obviously, for

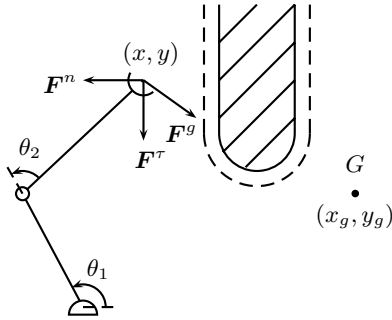


Fig. 1. R-R planar manipulator arm. The virtual forces produced by the tangent, the normal and the target sub-goals are shown. Note also the safety area around the obstacle

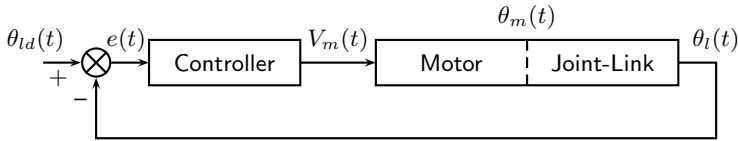


Fig. 2. Feedback control loop of a motorized joint-link pair

the execution of each joint angle's trajectory a conventional motorized feedback loop is needed, as shown in Fig. 2.

In summary, the control of the robotic arm can be efficiently resolved by means of conventional control techniques and, therefore, the essential and complex task that remains is to generate the appropriate trajectories of the links angles, which eventually leads to the obstacle avoidance problem.

3 Biomimetic Approach to Autonomous Robot Behavior

The design and development of robots endowed with as much autonomy as possible have been based either on hard computing techniques [6,7,8], with a strong emphasis on adaptation and learning, or soft computing techniques like fuzzy logic (FL) and artificial neural networks (ANN). Broadly speaking, both hard computing and FL methods, when applied to the design of autonomous robots, are based on *a priori* known formal representations or models of the environment and accurate knowledge of the robot's kinematics and dynamics. As for ANN methods, the aprioristic models of the environment are replaced by supervised trajectories that permit the robot to learn the appropriate actions.

In FL-based techniques for designing autonomous robots, the control actions of the robot are obtained by the designer from linguistic rules, and environment responses are also transformed into linguistic variables [9]. Therefore, the design is a process composed of intensive and extensive knowledge injected by the de-

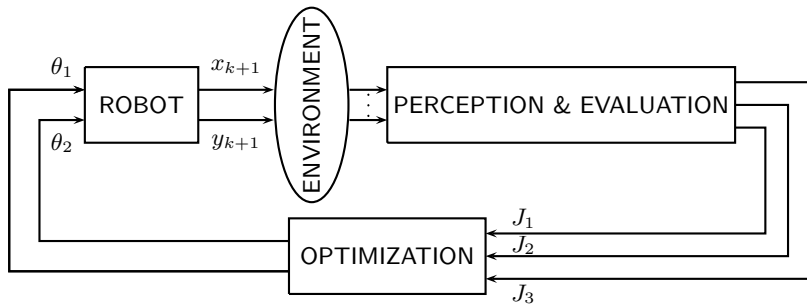


Fig. 3. Adaptation of the biomimetic method to the manipulator arm

signer into the final robotic system. As is well known, the design process and, particularly, the tuning of the membership functions of the linguistic variables is cumbersome and quite tricky [10]. An additional problem is the combinatorial explosion of the fuzzy control rules, particularly for dynamic and complex environments.

When using ANN techniques, the control actions of the robot are learnt by sensory-based experience and, more precisely, by following a teacher that provides the correct actions for each specific situation [11,12]. Although the ANN approach simplifies the design process substantially, the fact is that it calls for an exhaustive battery of training examples and, as is very well-known, the robotic system can get into serious trouble in the operating stage, if it comes up against situations for which it has not been trained.

Elsewhere we have introduced [13,14] a method based on the perception-reason-action cycle that is able to solve the navigation of robotic mechanisms without using formal representations of their interactions with unknown environments and without explicit knowledge of their kinematics. Let us now give an overview of the principles of this method.

4 Collision Avoidance

Fig. 3 shows the adaptation of our method to the particular case of an R-R planar manipulator arm. The key element of this biomimetic method is the optimization of the performance indices J_1 , J_2 and J_3 —to be explained later on—, which produces the robot actions or control variables, θ_1 and θ_2 . In fact, the definition of the performance indices is absolutely vital for the robot's global behavior, so that the design process revolves around the appropriate choice of such indices. Furthermore, these performance indices lie at the core of the ontological discussion about the degree of autonomy of any agent, including robots. We contend that no agent, be it natural or artificial, is truly autonomous as regards its ultimate intentions or goals —which, in the case of a robotic agent, are embodied in performance indices like the ones we are discussing—.

Our robot has at least two kinds of basic intentions or goals: to reach a final position or target and to avoid collisions with any existing obstacle. Many

performance indices can be defined from these two basic goals: to reach the target in minimum time or effort; to follow a straight line; to maintain a constant distance from the nearest obstacle, etc.

The attractiveness and strength of our method, as represented in the block-diagram of Fig. 3, is that the robot controller, as discussed above, needs neither explicit knowledge of the robot kinematics nor any formal model of the environment. The controller is exclusively based on an optimization process of the performance indices. As shown later on, this optimization is founded on the very simple, although powerful, idea of gradient-based search.

Sensu stricto, this method can be only considered as coordination of perception and action, without real reasoning in between. However, we shall consider two instances of emergent reasoning capabilities from the interaction between perception and action: (1) *if-then-else* rules that coordinate the different sub-goals of the robot and (2) formal representations acquired by the robot from its interaction with the environment.

5 Coordination of Goals: A Primitive Reasoning Ability

As mentioned above, the robot has two basic goals or intentions: to reach a target point within its working space and to avoid any interposed obstacle. Let (x, y) and (x_g, y_g) be the Cartesian coordinates of the robot end effector and the target point, respectively. Then, the goal of reaching the target can be expressed as the following performance index

$$J_1 = \frac{1}{2} [(x - x_g)^2 + (y - y_g)^2] \quad (1)$$

which must be minimized. The objective of reaching the final point (x_g, y_g) can be reformulated as a more restrictive objective of reaching the target by following a straight line trajectory: an astonishing human skill that requires a lot of injected knowledge when performed by a manipulator [15,16]. Apart from the use of the precise equations of the robot kinematics, the robot designer needs to enter the desired trajectory even for a simple R-R manipulator arm, like the one we are considering, to perform straight line trajectories. We are going to demonstrate that our biomimetic method performs the same task without using the kinematics equations and without injecting the desired trajectories into the robot controller. Instead, it suffices to inject an adequate performance index, which is:

$$J_1 = \frac{1}{2} \left[\tan^{-1} \frac{(y_k - y_{k-1})}{(x_k - x_{k-1})} - \tan^{-1} \frac{(y_g - y_{k-1})}{(x_g - x_{k-1})} \right]^2 \quad (2)$$

As for the second goal of obstacle avoidance, we define two sub-goals: (a) to maintain the end effector (x, y) outside a safety distance —see Fig. 1— and (b) to follow a trajectory as parallel to the nearest obstacle as possible. Elsewhere [17,18] we introduced a similar obstacle avoidance strategy for wheeled mobile robots, in which we defined two equivalent sub-goals: normal navigation

and tangential navigation. It can be shown that sub-goals (a) and (b) of the planar R-R robot are equivalent to normal navigation and tangential navigation, respectively. Normal navigation implies that the robot avoids entering the safety zone by a movement of the end effector which is normal to the nearest obstacle surface. Similarly, tangential navigation drives the end effector through a trajectory parallel to the obstacle surface. These two sub-goals can be materialized by the following two performance indices:

$$J_2 = \frac{1}{2} [d_{\min}(k) - d_s]^2 \quad ; \quad J_3 = \frac{1}{2} [d_{\min}(k) - d_{\min}(k-1)]^2 \quad (3)$$

where $d_{\min}(k)$ is the minimum distance of the end effector to the obstacle at instant k , and d_s is the safety distance. These distances can be easily measured by means of the motorized ultrasound-based sensor reported in [17]. Obviously, both indices J_2 and J_3 must be minimized. Index J_2 is only activated if $d_{\min} < d_s$; otherwise it is ignored. Index J_3 drives the end effector through a trajectory parallel to the obstacle contour. Fig. 1 shows the virtual forces created by the three performance indices.

The minimization of each individual performance index can be materialized [18] by a gradient descent algorithm:

$$\dot{\theta}_i^g = -\mu_i^g \frac{\partial J_1}{\partial \theta_i} \quad ; \quad \dot{\theta}_i^n = -\mu_i^n \frac{\partial J_2}{\partial \theta_i} \quad ; \quad \dot{\theta}_i^\tau = -\mu_i^\tau \frac{\partial J_3}{\partial \theta_i} \quad (4)$$

where θ_i^g , θ_i^n and θ_i^τ are the robot actions generated by each individual sub-goal. The discrete-time version for, say, the first sub-goal is

$$\theta_i^g(k+1) = \theta_i^g(k) - \mu_i^g \left. \frac{\partial J_1}{\partial \theta_i} \right|_{\theta_i(k)} \quad ; \quad i = 1, 2 \quad (5)$$

Each individual action depends on the sensory-based gradients, which are computed using the measurements provided by the robot sensors. Due to this sensory dependency, the modules of the respective control outputs tend to be disproportionate to the internal dynamics of the robot: sometimes the sensory-based gradients are too strong or, inversely, they are too small, as far as the robot's natural dynamics is concerned. To solve this problem of scale, we have devised a look-up table mapping the sensory-based gradients module to the natural dynamics of the robot. Note that this mapping is related to the distinction made by psychologists between the distal space of sensations and the proximal space of actions [16].

As for the final control action of the robot, it must be formed by a combination of the actions recommended by each sub-goal:

$$\theta_i(k+1) = \omega_{i1}\theta_i^g(k+1) + \omega_{i2}\theta_i^n(k+1) + \omega_{i3}\theta_i^\tau(k+1) \quad ; \quad i = 1, 2 \quad (6)$$

If we interpret each individual sub-goal as the robot's basic behaviors, then the crucial issue is the appropriate coordination of these behaviors. A robot that does not adequately coordinate its behaviors will have a lot of trouble navigating

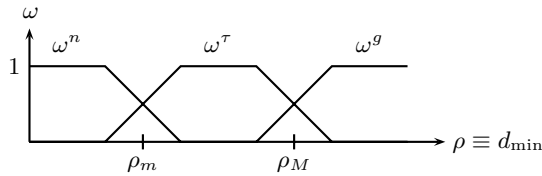


Fig. 4. Distribution of the coordination parameters as a function of the variable ρ (see the text for an explanation)

through unknown environments. Consequently, the basic coordination of perception and action performed for each individual behavior will not be sufficient for successful navigation. Therefore, the robot must introduce some intermediate activity between its perceptions —basically distances— and its actions —i.e. motorized movements of its joint angles—. This intermediate activity, which basically consists of a coordination of the reactive behaviors, can be considered as a primitive reasoning capability.

Looking at expression (6), we find that the coordination of the sub-goals is based on the setting of the parameters ω_{ij} ($i = 1, 2; j = 1, 2, 3$) that weight the importance of each sub-goal. In this respect, a simple reasoning process could be based on something like the following linguistic rules:

1. “*if the robot is near to the obstacle, then sub-goal J_2 is critical*”
2. “*if the robot is far from the obstacle, then sub-goal J_1 should be dominant*”
3. “*if the robot is at an intermediate distance, then a tangential movement is recommendable, as far as a smooth and elegant trajectory for collision avoidance is concerned*”

This reasoning expressed as linguistic rules can be formalized and materialized either by a rather complex fuzzy logic-based controller or by a much simpler approach based on the dynamic coordination of multi-objective performance indices [13,14,17]. Applying the latter approach, first we introduce a normalization constraint on the coordination parameters

$$\sum_{j=1}^3 \omega_{ij} = 1 \quad ; \quad i = 1, 2 \quad ; \quad 0 \leq \omega_{ij} \leq 1 \quad (7)$$

The above reasoning rules can be transformed into the distributions of the coordination parameters shown in Fig. 4, where we have two critical parameters ρ_m and ρ_M .

For simplicity's sake, we have made two slight changes to the notation. First, we have dropped the subindices i and j of the coordination parameters to emphasize the respective sub-goal: normal, tangent and target or goal. Second, we have substituted $d_{\min}$ (i.e. the minimum radial distance from the end effector to the nearest obstacle) by ρ , to avoid additional subindices in the notation. In Fig. 4, the critical parameters ρ_m and ρ_M quantify the words *near* and *far*

appearing in the reasoning rules, respectively. The interval $\rho_m - \rho_M$ determines the linguistic expression *intermediate distance*. Obviously, the shape of distribution of the coordination weights ω^n , ω^τ and ω^g may vary and is not necessarily trapezoidal as shown in Fig. 4

Although the correct choice of the critical parameters ρ_m and ρ_M is an open problem, which can be tackled using different methods like evolutionary and genetic programming or by conventional gradient-based optimization, they have been settled in this paper by means of a trial-and-error design process. We will return to this topic later on, in the section explaining the experimental results. Let us now continue with a discussion of the role played by a formal representation of the environment, autonomously acquired by the robot, and that can be considered as an additional reasoning capacity.

6 Model-Based and Model-Free Robot Behaviors

As mentioned above, the dynamic acquisition by the robot of a formal representation or model of its interaction with the environment is clearly a reasoning ability, which allows the robot to plan its future actions and even to simulate in advance the effects of its possible actions on the state of the environment.

By model-free we understand the situation in which the robot decisions — i.e., its control actions θ_1 and θ_2 — rely on the empirical and instantaneous information gathered by its sensors:

$$\left. \frac{\partial J_i}{\partial \theta_j} \right|_{\theta_j(k)} = \frac{J_i(k) - J_i(k-1)}{\theta_j(k) - \theta_j(k-1)} \quad ; \quad i = 1, 2, 3 \ ; \ j = 1, 2 \quad (8)$$

Therefore, by computing the estimated gradient of the sensory-based performance indices $J_i(\theta_1, \theta_2)$, the robot is behaving as a model-free agent, embedded in a dynamic environment. Such a sensory-based behavior can be interpreted as purely reactive, in the sense that perception and action are directly coupled, without any type of intermediate processing. However, this simple and model-free behavior is extremely powerful, as we find by looking at the experimental results shown in Fig. 7, in which the robot has successfully performed several locomotion tasks.

The secret of the excellent performance achieved by the robot lies in two elements: (1) the objective functions evaluating its performance, which are in turn based on sensory information, and (2) the action strategy based on the optimization of the performance functions. Furnished with these two elements, the robot is able to interact with its environment without using a formal representation or model of its interaction with the environment, and without making explicit use of its own kinematics — i.e., the transformation between its internal coordinates and the environment coordinates —.

Let us now investigate the use of formal representations of the robot-environment ensemble. We emphasize that we are talking about modelling the joint dynamics of the agent and its particular environment, rather than modelling each

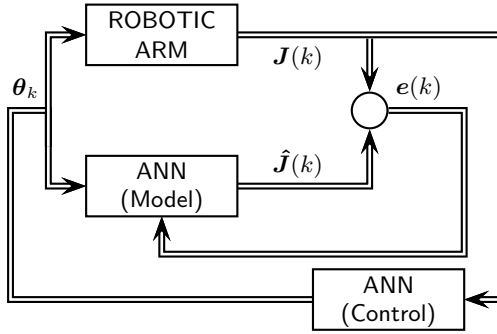


Fig. 5. Conceptual block-diagram of the modelling process using an ANN. Note that the control actions are generated by another ANN minimizing the performance functions

part individually. Then, we are going to evaluate robot performance when its behavior is based on a formal representation or model of its interaction with the environment. Fig. 5 represents the modelling process undertaken by the manipulator by means of an ANN. Note that the model built by the ANN represents the relationship between the robot actions and the responses of the environment, as quantified by performance indices. This ANN has been implemented via a multilayer perceptron with backpropagation of the model error.

It should be noted that we have explicitly introduced a second ANN to produce the control actions. This ANN, as shown in Fig. 5, is based on the same idea of utilizing the empirically estimated gradients of the indices and it has been implemented by means of a simple perceptron. Then, we can conclude that the agent's formal representation of its interaction with the environment is not used for control purposes, as the joint angles are exclusively based on the gradient of the performance functions obtained from the sensor readings. However, once an accurate model of the manipulator-environment ensemble has been built by the respective ANN, then the agent actions can also be generated by the model. This kind of model-based control is known as indirect control, as it is obtained from a model of the plant under control [16]. Fig. 6 shows the block-diagram of an hybrid control architecture, in which we have simultaneously considered both type of control actions: (1) model-free or direct control and (2) model-based or indirect control. Note also the ANN in charge of the tricky task of properly combining both control actions. Superindices e and m stand for empirical or direct and model, respectively.

7 Experimental Results

To demonstrate the effectiveness of the proposed method for the autonomous locomotion of a planar R-R manipulator, we present in Fig. 7, a specific situation in which there are unknown obstacles interposed between the end effector starting position and the target location.

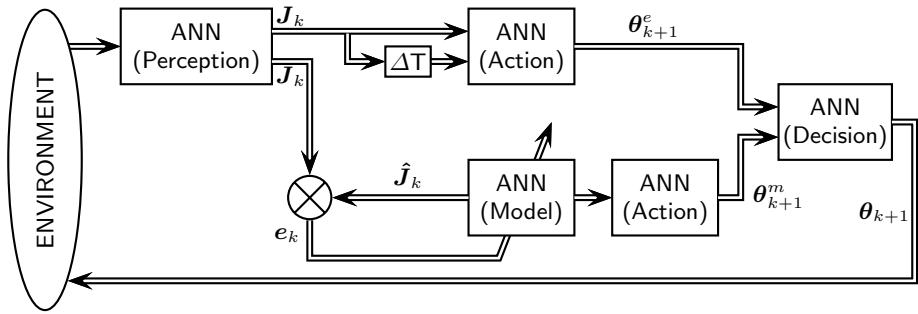


Fig. 6. Block-diagram of the combined direct and indirect control actions. Note the vectors representing more than just one signal or function

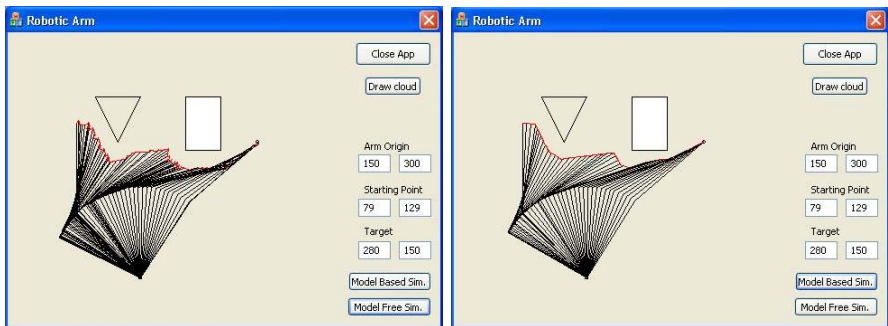


Fig. 7. Navigation of the manipulator arm acting as a model-free agent in (a) and as a model-based agent (b)

In these simulations, we have evaluated and compared the performance achieved by the manipulator arm with both model-free and model-based implementations, using the same coordination policy described in the paper in each case. As shown in Fig. 7, the manipulator starts to navigate towards the target following a straight line until it detects some of the existing obstacles. At this point, the behaviors corresponding to normal navigation and tangential navigation are activated. When the robot reaches a region in which there is an open space in front of the target position, the robot abandons the two collision avoidance sub-goals in favor of the target sub-goal, following a straight-line trajectory again.

As regards the performance of both the model-free and the model-based implementations, the improvement in the smoothness of the manipulator trajectories when using a formal representation of its environment, dynamically obtained by the ANN inside the robot controller, is particularly noteworthy. However, this improvement is accomplished at the cost of a more complex design process, as tuning the ANN in charge of the modelling task is a tricky and cumbersome process.

8 Concluding Remarks

A robotic mechanism that is able to acquire primitive reasoning abilities from the coordination of perception and action has been presented. The robot structure is materialized by means of a bio-inspired architecture, in which perceptions from the environment and robot actions manage to autonomously coordinate themselves, chiefly through environment responses. More specifically, we have considered the hard problem of dynamic collision avoidance —i.e., the type of task that for a conventional robot requires complete, a priori knowledge of the environment— as the test-bed for our robot. We have demonstrated in practice two interesting features of our approach as regards the conventional, model-based approach: (1) the robot actions are performed without solving the kinematics equations and (2) there is no need for the robot to have a formal representation of the environment in order to negotiate complex environments and avoid obstacles.

We have also investigated the possibilities for the robot to acquire a simple reasoning ability by injecting an additional structure into the robot controller to embody simple, although powerful, *if-then-else* rules. This enables the robot to transcend its previous reactive behaviors based on pure interaction between perception and action, which can be considered as a primitive reasoning activity of a sort. Learning new strategies for obstacle avoidance illustrate the potentiality added to the robot by such reasoning ability.

Acknowledgements. This work has been partially supported by the Spanish Ministry of Science and Technology, project DPI2002-04064-CO5-05. We also thank Javier Alonso-Ruiz for his contribution to the experimental work.

References

1. Scheier, C., Pfeifer, R. (1999) The embodied cognitive science approach. In W. Tschacher, J.-P. Dauwelder (eds.) *Dynamics Synergetics Autonomous Agents*. World Scientific, Singapore, 159–179
2. Kortenkamp, D., Bonasso, R.P., Murphy, R. (1998) *Artificial Intelligence and Mobile Robots*. The AAAI Press/The MIT Press, Cambridge, Massachusetts
3. Mira, J., Delgado, A.E. (2003) Where is knowledge in robotics? Some methodological issues on symbolic and connectionist perspectives of AI. In C. Zhou, D. Maravall, D. Ruan (eds.) *Autonomous Robotic Systems: Soft Computing and Hard Computing Methodologies and Applications*. Physica-Verlag, Springer, Heidelberg, 3–34
4. Fodor, J., Pylyshyn Z. (1988) Connectionism and cognitive architecture: A critical analysis. *Cognition*, **28**, 3–71
5. Van Gelder, T. (1997) The dynamical alternative. In D.M. Johnson, C.E. Erneling (eds.) *The Future of the Cognitive Revolution*. Oxford University Press, Oxford, 227–244
6. Latombe, J.-C. (1995) Robot algorithms. In K. Goldberg, D. Halperis, J.-C. Latombe, R. Wilson (eds.) *Algorithmic Foundations of Robotics*. A.K. Peters, Wellesley, Massachusetts, 1–18

7. Zhou, C., Maravall, D., Ruan, D. (2003) Autonomous Robotic Systems: Soft Computing and Hard Computing Methodologies and Applications. Physica-Verlag, Springer, Heidelberg
8. Siciliano, B. (2001) Robot control. In T. Samad (ed.) Perspectives in Control Engineering. IEEE Press, New York, 442–461
9. Mendel, J.M. (1999) Fuzzy logic systems for engineering: A tutorial. Proceedings of the IEEE, **83**(3), 345–377
10. Hitchings, M., Vlacic, L., Kecman, V. (2001) Fuzzy control. In L. Vlacic, M. Parent, F. Harashima (eds.) Intelligent Vehicle Technologies. Butterworth & Heinemann, Oxford, 289–331
11. Kong, S.G., Kosko, B. (1992) Comparison of fuzzy and neural track backer-upper control systems. In B. Kosko (ed.) Neural Networks and Fuzzy Systems, Prentice-Hall, Englewood Cliffs, New Jersey, 339–361
12. Lewis, F.L., Jagannathan, S., Yesildirek, A. (1999) Neural Network Control of Robot Manipulators and Nonlinear Systems. Taylor & Francis, London
13. Maravall, D., de Lope, J. (2002) A reinforcement learning method for dynamic obstacle avoidance in robotic mechanisms. In D. Ruan, P. D'Hondt, E.E. Kerre (eds.). World Scientific, Singapore, 485–494
14. Maravall, D., de Lope, J. (2003) A bio-inspired robotic mechanism for autonomous locomotion in unconventional environments. In C. Zhou, D. Maravall, D. Ruan (eds.) Autonomous Robotic Systems: Soft Computing and Hard Computing Methodologies and Applications. Physica-Verlag, Springer, Heidelberg, 263–292
15. Kawato, M. (2003) Cerebellum and motor control. In M.A. Arbib (ed.) The Handbook of Brain Theory and Neural Networks, 2nd edition. The MIT Press, Cambridge, Massachusetts, 190–195
16. Jordan, M.I., Rumelhart, D.E. (1992) Forward models: Supervised learning with a distal teacher. Cognitive Science, **16**, 307–354
17. De Lope, J., Maravall, D. (2003) Integration of reactive utilitarian navigation and topological modeling. In In C. Zhou, D. Maravall, D. Ruan (eds.) Autonomous Robotic Systems: Soft Computing and Hard Computing Methodologies and Applications. Physica-Verlag, Springer, Heidelberg, 103–139
18. Maravall, D., de Lope, J. (2003) Integration of artificial potential field theory and sensory-based search in autonomous navigation. Proc. of the IFAC 2002 World Congress, Elsevier (to appear)

Inverse Kinematics for Humanoid Robots Using Artificial Neural Networks

Javier de Lope, Rafaela González-Careaga, Telmo Zarraonandia, and
Darío Maravall

Department of Artificial Intelligence
Faculty of Computer Science
Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Madrid, Spain
`{jldlope,rafaela,telmoz,dmaravall}@dia.fi.upm.es`

Abstract. The area of inverse kinematics of robots, mainly manipulators, has been widely researched, and several solutions exist. The solutions provided by analytical methods are specific to a particular robot configuration and are not applicable to other robots. Apart from this drawback, legged robots are inherently redundant because they need to have real humanoid configurations. This degree of redundancy makes the development of an analytical solution for the inverse kinematics practically unapproachable. For this reason, our proposed method considers the use of artificial neural networks to solve the inverse kinematics of the articulated chain that represents the robot's legs. Since the robot should always remain stable and never fall, the learning set presented to the artificial neural network can be conveniently filtered to eliminate the undesired robot configurations and reduce the training process complexity.

1 Introduction

The design and development of legged robots has been one of the main topics in advanced robotic research. Basically, projects addressing legged robots study the stability and mobility of these mechanisms in a range of environmental conditions for applications where wheeled robots are unsuitable. The research in this field is been oriented to the development of humanoid robots. There are successful projects, such as the Honda, Waseda and MIT humanoid robots.

The main interest has focused on getting the robot to maintain stability as it walks in a straight line. Two methods are usually used for this purpose. The first and most widely used method aims to maintain the projection of the center of masses (COM) of the robot inside the area inscribed by the feet that are in contact with the ground. It is known as *static balance*. The second method, also referred to as *dynamic balance*, uses the Zero Moment Point (ZMP), which is defined as the point on the ground around which the sum of all the moments of the active forces equal zero [1]. If the ZMP is within the convex hull of all contact points between the feet and the ground, the biped robot is stable and will not fall over [2,3].

This paper proposes a method for obtaining relative robot foot positions and orientations to achieve several sets of postures, apart from walking in a straight line, like turning around, moving sideways, stepping backwards, etc. The standing and walking postural schemes are discussed.

To achieve these postures, we must give the new foot coordinates to the control system, and it will be able to determine the way the leg should move in order to place the foot at the new position and orientation. For this purpose it will be necessary to compute the inverse kinematics of the legged robot.

A direct kinematic model of the system will have to be built to provide the artificial neural network with a set of training cases. Because of the redundancy of the degrees of freedom (DOF), one feet position can map several angle configurations and a stability criterion will have to be applied to determine the best suited configuration.

Several works have been reported on training neural networks to resolve the inverse kinematics of a robot arm. Self-organizing maps have been widely used for encoding the sample data on a network of nodes to preserve neighborhoods. Martinetz *et al.* [4] propose a method to be applied to manipulators with no redundant DOF or with a redundancy resolved at training time, where only a single solution along a single branch is available at run time. Jordan and Rumelhart [5] suggest first training a network to model the forward kinematics and then prepending another network to learn the identity, without modification of the weights of the forward model. The prepended network is able to learn the inverse. Other alternatives, like recurrent neural networks [6], are also used to learn the inverse kinematics of redundant manipulators.

Kurematsu *et al.* [7] use a Hopfield-type neural network to solve the inverse kinematics of a simplified biped robot. The joint positions are obtained from the position of the center of gravity and the position of the toes is calculated from the equation of an inverted pendulum.

2 Mechanical Description of the Humanoid Robot

Currently a real robot is been designed and constructed. Light weight materials, such as aluminium, are being used. The joints are driven by servo modules for radio controlled models with a limited and reduced torque, which are not applied directly over the joint to increase servo motor performance.

At this stage of research, we are considering a simplified humanoid robot which is made up two legs and the hip. So, we are considering a biped robot. Each leg is composed of six degrees of freedom, which are distributed as follows: two for the ankle—one rotational on the pitch axis and the other one rotational on the roll axis—, one for the knee—rotational on the pitch axis—, and three for the hip—each of them rotational on each of the axes—.

The ankle joint is one of most complex joints in a biped robot. It integrates two rotations of the foot in the roll and pitch axes. The former rotation moves the leg forward during a step and the latter rotation is used, in conjunction with a lateral rotation of the hip around its roll axis, to balance the body and adjust

the position of the COM projection. Therefore, two servos are required to rotate the foot around the roll and pitch axes.

The servos are usually situated in the shank, rotated 90 degrees with respect to the other to allow both movements as, for example, in the PINO [8] and Robo-Erectus [9] humanoids. This configuration is not very well suited because the movement produced by the leg as a whole is not like human movement. With this arrangement of the axes, the pitch rotation is made immediately above the foot, but the roll rotation takes place in the middle of the shank. The human ankle can produce both rotations at the same point, so another kind of mechanism must be used/designed for making more natural, human movements.

A possible solution is provided by humanoid robots like P3 [10], BIP [11] or HRP [12]. These robots use a double axis joint by means of a cardan and two parallel motors on the shank. The axis of every motor is fixed to the foot and prismatic movements of the motors mean the foot can rotate around the pitch and roll axes.

This solution can also be adopted using servos. Two servos can be mounted on the shank with the rotation axes at right angles to the shank and parallel to the foot plane. The two rotations of the foot can be achieved by actuating both servos simultaneously. This joint, which is referred to as *cardan-type joint*, is briefly studied in the next section.

The knee joint only includes a rotation around the pitch axis. In this case, the servo can directly drive the joint, but the required servo torque can be reduced if we consider the use of a simple lever and the mechanical advantage that it provides. Adjusting the position of the fulcrum of the lever on the leg, we will reduce both the required torque and the range of movement of the joint. The *lever-type joint* will be discussed in the following.

Three different rotations must be contemplated for the hip joint, around each one of the rotation axes. These rotations can be divided into two groups: a single rotation around the pitch axis, similar to the knee joint, and a double rotation around the roll and yaw axes. The single rotation needs more torque, because it has to lift up the entire leg by a greater range and can be implemented as a lever-type joint. We can use a cardan-type joint for the double rotation, as required ranges and torques for movements are lower.

2.1 Cardan-Type Joint

Fig. 1 shows a cardan-type joint driven by two servos. Let α and α' be the angular position of the right and left servos, respectively. It is known that the range of positions for these angles in servos for radio controlled models is 180 degrees, approximately. Moving the servos in the same direction, clockwise or counterclockwise, the contiguous link rotates around the roll axis, because L grows at the same rate as L' decreases and vice versa. When the servos are moved in opposite direction, the contiguous link rotates around the pitch axis, because L and L' grow or decrease at the same rate.

Let us consider just the right-hand servo. L_1 is the servo arm. One end of the segment L_2 is fixed to the servo arm and the other end to the contiguous

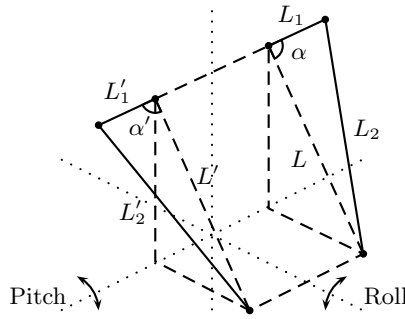


Fig. 1. Rotation axes of a cardan-type joint

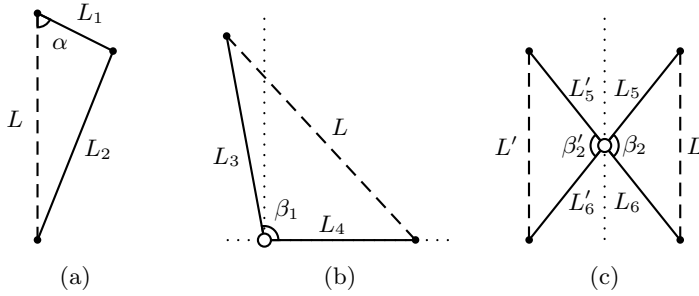


Fig. 2. The ankle joint and its parameters

link. In this way, as L_1 and L_2 are rigid, the length of L is determined by the servo angle α (see Fig. 2a) according to the following expression:

$$L_2^2 = L_1^2 + L^2 - 2L_1L \cos \alpha \quad (1)$$

and L can be expressed as function of the other parameters, which are known or must be estimated during the design phase

$$L = L_1 \cos \alpha \pm \sqrt{L_1^2 \cos^2 \alpha - L_1^2 + L_2^2} \quad (2)$$

If the servos are moved in opposite directions, the value of L and L' will be the same and will grow or decrease according to the value of the angles α and α' . Therefore, by increasing or decreasing the length of L (and L'), the contiguous link will experiment a pitch rotation around the axis. The magnitude of this rotation is denoted as β_1 in Fig. 2b. L_3 is the distance between the servo and pitch axes. L_4 determines the distance between the pitch axis and the fixing point. By modifying the length of all these parameters, we can determine the movement range around the pitch axis.

The parameter β_1 is determined by the expression:

$$L^2 = L_3^2 + L_4^2 - 2L_3L_4 \cos \beta_1 \quad (3)$$

By operating, we get the value of β_1 :

$$\beta_1 = \arccos \frac{L_3^2 + L_4^2 - L^2}{2L_3L_4} \quad (4)$$

where L is determined by (2).

Now, let us consider the rotation around the roll axis. This movement is produced when the servos move in the same direction, and L and L' increase or decrease at the same rate (if L increases, L' decreases and vice versa). L can also be expressed as follows:

$$L^2 = L_5^2 + L_6^2 - 2L_5L_6 \cos \beta_2 \quad (5)$$

where β_2 is the rotation angle around the roll axis, L_5 is the distance between the servo and roll axes, and L_6 is the distance between the roll axis and the fixing point, as shown in the Fig. 2c.

By operating, we get the value of β_2 as a function of the other parameters:

$$\beta_2 = \arccos \frac{L_5^2 + L_6^2 - L^2}{2L_5L_6} \quad (6)$$

where L is determined by (2).

Note that β_2 and β'_2 are two *complementary* angles in this kind of movement. We have assumed that when a servo rotates clockwise, the other one also rotates clockwise, so when β_2 increases, β'_2 decreases and the rotation around the roll axis is produced. The range of movement of β_2 and β'_2 to the left and right is different, if the same interval is taken for α and α' . Obviously, due to mechanical and physical restrictions, the minimum of both β_2 angles must be selected in order to determine the range around the roll axis.

As we are using a cardan joint, the roll axis cut the pitch axis at a point at which both rotations are produced. Accordingly, we can assume that $L_3 = L_5$ and $L_4 = L_6$. If the axes are crossed, but not cut, for example, when two different axes are used rather than a cardan, this assumption is not valid.

To get the range of movement around each axis, all the L_i parameters must be fixed.

2.2 Lever-Type Joint

Fig. 3 shows a simplified diagram of the lever-type joint and the used parameters for the computation of the joint range. L_1 is the length of the servo arm, L_2 is the length of a rigid segment that connects the servo arm and the contiguous link, L_3 is the distance between the servo and joint axes, and L_4 is the distance between the rotation axis and the point at which the rigid segment is fixed to the contiguous link. α is the rotation angle of the servo and, as mentioned above, it is constrained approximately to a range of 180 degrees. The angle $\gamma = \gamma_1 + \gamma_2$ defines the joint position and its range of values can be calculated from the L_2 , L_3 and L_4 parameters, and it will be selected considering the most appropriate range for the required movements and the generated force/torque.

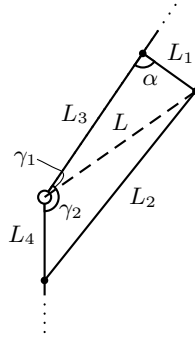


Fig. 3. The lever-type joint and its parameters

The length of the segment L is determined by the following expression:

$$L^2 = L_1^2 + L_3^2 - 2L_1L_3 \cos \alpha \quad (7)$$

L_1 and L_2 parameters can be expressed as:

$$L_1^2 = L^2 + L_3^2 - 2LL_3 \cos \gamma_1 \quad (8)$$

$$L_2^2 = L^2 + L_4^2 - 2LL_4 \cos \gamma_2 \quad (9)$$

where L is determined by (7).

By operating, we get the value of γ by means of:

$$\gamma = \gamma_1 + \gamma_2 = \arccos \frac{L^2 + L_3^2 - L_1^2}{2LL_3} + \arccos \frac{L^2 + L_4^2 - L_2^2}{2LL_4} \quad (10)$$

For the construction point of view, the lengths of L_3 and L_4 are limited by the length of the links. Moreover, the length L_4 is, the larger the distance between L_2 and the leg is. If this distance is too large, the workspace needed for the leg will also be larger and troubles when the joint is completely bent could exist.

On the other hand, we can also consider some dynamical aspects related to the moment arm and the servo torque. Assuming that the link is almost perpendicular to the ground most of the time during operation, we can choose a value for L_4 which establish an advantageous moment arm to compensate a lower torque rating of the servo. The mechanical advantage is defined by the L_1 and L_4 lengths considering the whole system as a lever (see the Fig. 3). For example, considering a value for $L_4 = 2L_1$, we can approximate that the requiring torque ratio for the servo to a half.

2.3 Physical Dimensions

Table 1 shows the maximum ranges for each robot joint. θ_1 and θ_2 correspond to the pitch and roll axes of the right ankle (θ_{12} and θ_{11} , for the left ankle). θ_3

Table 1. Range of angle positions for each joint

<i>Right Leg</i>			<i>Left Leg</i>		
θ_i	<i>Min</i>	<i>Max</i>	θ_i	<i>Min</i>	<i>Max</i>
θ_1	-42.55°	27.82°	θ_7	-35.31°	32.21°
θ_2	-27.82°	27.82°	θ_8	-32.21°	32.21°
θ_3	-130°	0°	θ_9	-14.73°	104.43°
θ_4	-104.43°	14.73°	θ_{10}	0°	130°
θ_5	-32.21°	32.21°	θ_{11}	-27.82°	27.82°
θ_6	-35.31°	32.21°	θ_{12}	-27.82°	42.55°

and θ_{10} are the angles associated to the right and left knees, respectively. The rest of angles are belonged to the hips.

The total height of the robot is 274 mm. The links dimensions are 10, 106 and 158 mm for the links between the ground and the ankle, the ankle and the knee, and the knee and the hip respectively. The hip length is 100 mm.

3 Direct Kinematics

The direct kinematics of the robot as regards the relative position and orientation of one foot from the other can be solved easily considering our model as a robotic chain of links interconnected to one another by joints. The first link, the base coordinate frame, is the right foot of the robot. We assume it to be fixed to the ground for a given final position or movement, where the robot's global coordinate frame will be placed. The last link is the left foot, which will be free to move. The assignment of the coordinate frames to the robot joints is illustrated in Fig. 4.

Obviously, to consider the positions or movements in the inverse case, that is, when the left foot is fixed to the ground and the right foot is able to move, the coordinate frames must be assigned similarly. This does not lead to any important conclusions concerning the proposed method and will not be discussed in this paper.

The position and orientation of the left with respect to the right foot will be denoted by the homogeneous coordinate transformation matrix:

$${}^R T_L = \begin{bmatrix} {}^R R_L & {}^R \mathbf{p}_L \\ 0 & 1 \end{bmatrix} = {}^R T_0 \times {}^0 T_1(\theta_1) \times \dots \times {}^5 T_6(\theta_6) \times {}^6 T_7(\theta_7) \times \dots \times {}^{11} T_{12}(\theta_{12}) \times {}^{12} T_L \quad (11)$$

where ${}^R R_L$ denotes the rotation matrix of the left foot with respect to the right foot coordinate frame, ${}^R \mathbf{p}_L$ denotes the position matrix of the left foot with respect to the right foot coordinate frame and each ${}^i T_j$ denotes the homogeneous coordinate transformation matrix from link i to link j . Coordinate frames from 1 to 6 correspond to the right leg, and from 7 to 12 are associated to the left leg.

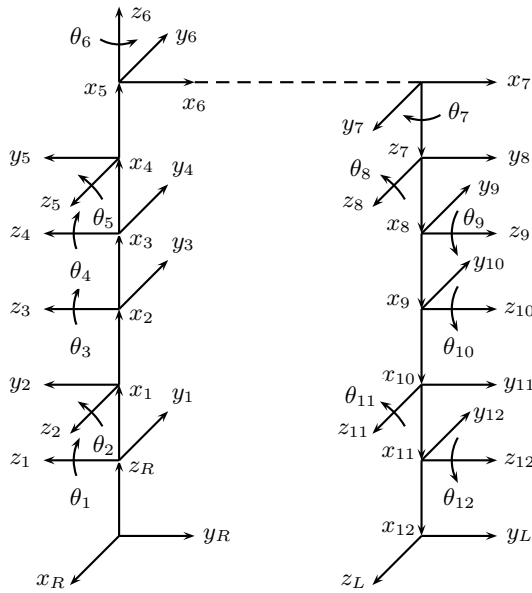


Fig. 4. Coordinate frames associated to the robot joints

4 Postural Schemes

Based on the model of the robot and the direct kinematics described above, a set of joint and Cartesian coordinates pairs has been generated to be used as training data for the neural network. Two sets of postural schemes have been considered. The first one will be composed of positions with both feet on the ground, straightened legs, which are uniformly distributed across a given coordinate range. The second scheme includes the positions involved in the execution of a single robot step.

For each postural scheme, a different artificial neural network will be used, and the respective network will be selected depending on the posture to be achieved. Each artificial neural network will be trained separately with the respective generated data under the restrictions explained below.

For each case, the redundancy of the degrees of freedom will mean that different joint configurations can produce the same feet position. Therefore, the criteria we will use to choose which configurations will be used as training data need to be established. Ideally, we want the network to learn the most stable positions from the range of all the possible solutions. Using a static balance criterion, a position is considered to be more stable the closer the vertical projection of the COM of the humanoid is to the center of the support polygon formed by the feet in contact with the ground. The height of the COM will also affect system stability, where the system will be more stable the lower it is placed. However, we are not interested in postures that place the hip too low, as we want the postures of the robot to be as natural as possible.

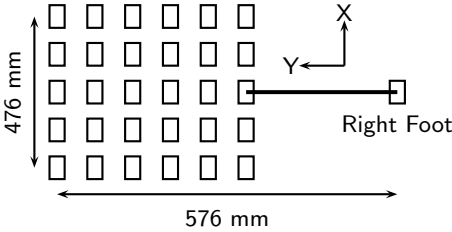


Fig. 5. Positions of the free foot in the training set of the standing postural scheme

Table 2. Range of angle positions for the standing postural scheme

<i>Right Leg</i>			<i>Left Leg</i>		
θ_i	<i>Min</i>	<i>Max</i>	θ_i	<i>Min</i>	<i>Max</i>
θ_1	-30°	25°	θ_7	-30°	30°
θ_2	-25°	0°	θ_8	0°	30°
θ_3	-90°	0°	θ_9	-12°	30°
θ_4	-30°	12°	θ_{10}	0°	90°
θ_5	0°	30°	θ_{11}	0°	25°
θ_6	-30°	30°	θ_{12}	-25°	30°

4.1 Standing Postural Scheme

The aim of this first scheme is to learn robot postures along the roll and pitch axes. These postures are restricted to both feet being in full contact with the ground, the knees not being bent and there being no rotation around the hip roll axes. Fig. 5 shows the final positions of the free foot.

A set of joint and Cartesian coordinates pairs whose values fulfill these constraints will be generated to feed the neural network. The training set is composed of 1000 pairs. The range of angle positions is shown in Table 2.

A two-layer backpropagation network is used to learn a representative set of input-target pairs, which has been introduced in elsewhere [13]. The input layer has 3 neurons, one for each element of the Cartesian coordinates of the free foot. The output layer has 12 neurons, associated to the joint coordinates of the humanoid robot. After trying out several configurations, the hidden layer contains 35 neurons.

The transfer function for the output layer has to be a *purelin* function, because the outputs can take any value. The *logsig* transfer function was chosen for the hidden layer because it achieves a lower error rate than the other transfer functions compared. The Levenberg-Marquardt training algorithm was the fastest training algorithm and 32 epochs were needed to reach the error goal 0.001. By comparison, the epochs needed to reach an error goal of 0.01, ten times greater than the above, using, for example, the variable learning rate algorithm, is six orders of magnitude.

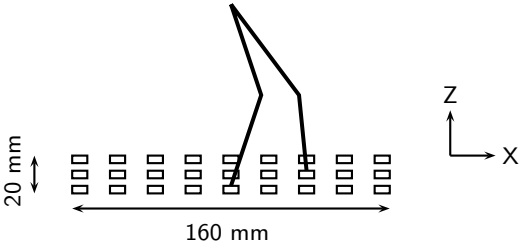


Fig. 6. Positions of the free foot in the training set of the walking postural scheme

Table 3. Range of angle positions for the walking postural scheme

<i>Right Leg</i>			<i>Left Leg</i>		
θ_i	<i>Min</i>	<i>Max</i>	θ_i	<i>Min</i>	<i>Max</i>
θ_1	0°	25°	θ_7	—	—
θ_2	−15°	15°	θ_8	−15°	15°
θ_3	−30°	0°	θ_9	0°	30°
θ_4	−30°	0°	θ_{10}	0°	30°
θ_5	−15°	15°	θ_{11}	−15°	15°
θ_6	—	—	θ_{12}	−25°	0°

4.2 Walking Postural Scheme

The aim of the second scheme is to learn all the robot postures required to take a step on the roll axis. For this to be achieved, we have to generate a set of joint and Cartesian coordinates pairs which describes possible postures of the robot to be adopted along the step gait (Fig. 6).

Any of these postures must accomplish the COM stability criteria, and for this second scheme we can make a distinction between two different cases:

- (a) The robot has only one foot in contact with ground. In this case the polygon of support is formed only by the right foot, and the projection of the COM must remain as close as possible to the center of this rectangle. For this to be achieved, some amount of swinging movement of the hip will be necessary.
- (b) The robot has both feet in full contact with the ground. Here the polygon of support will be the one formed by the two feet. In order to determine its shape, we calculate the convex hull of the polygon [14]. To calculate the center position of the polygon, we get the arithmetic average of the cut points of the lines perpendicular to the polygon sides.

The process for outputting the data is to generate a set of values that uniformly covers all the angle ranges of the roll and pitch axes. The values of the angles for the roll joints of the right hip and the left ankle are calculated to assure that the hip is parallel to the ground and that the left foot always remains parallel to the floor. Table 3 shows the used ranges.

Another set of angles must be generated to obtain the desired balancing movement of the hip and maintain the COM projection inside the support polygon. These values will be applied appropriately so that the pitch axes of the ankle and hip joints of both legs maintain the hip parallel to the ground.

Once we have the set of values for each angle, we generate all the possible combinations, calculate the final position of the left foot, the position of the COM and apply the restrictions described above.

The next filter to be applied, removes the less stable repeated positions (the ones with the projection of the COM further from the center of the polygon), leaving unique and optimally stable positions to train the neural network.

Finally, we restrict the length and height of the step. The movement ranges for every axis are ± 80 mm in the roll axis, ± 8 mm in the pitch axis and 20 mm for the maximum high of the feet over the ground [2].

The result is the final set of joint and Cartesian coordinates pairs that will be used to train the neural network. This set is composed of 300 input-target pairs, approximately.

The neural network for this postural scheme has the same architecture as the first artificial neural network described. Nevertheless, as the complexity of the input data have increased considerably, the number of neurons in the hidden layer has been increased to 100.

5 Conclusions and Further Work

A method for solving the inverse kinematics of a humanoid robot has been presented. The method is based on artificial neural networks and makes it unnecessary to develop an analytical solution for the inverse problem, which is practically unapproachable because of the redundancy in the robot joints.

An artificial neural network learns relative foot positions and orientations to achieve several postures. Each artificial neural network is trained separately with the generated data under a set of constraints and stability criteria.

The proposed neural architecture is able to approximate, at a low error rate, the learning samples and to generalise this knowledge to new robot postures. The error in the roll, pitch and yaw axes are close to 1 millimeter, less than the 1% considering the leg height.

The next stage of our work will be to define a mechanism for the selection of the most appropriate postural schemes for each robot movement. Also, the generation of other postural schemes apart from the standing and walking postural schemes will be implemented.

Acknowledgements. This has been partially supported by the Spanish Ministry of Science and Technology, project DPI2002-04064-CO5-05.

References

1. Vukobratovic M., Brovac, B., Surla, D., Sotkic, D. (1990) Biped Locomotion. Springer-Verlag, Berlin
2. Huang, Q. *et al.* (2001) Planning walking patterns for a biped robot. IEEE Trans. on Robotics and Automation, **17**(3), 280–289
3. Furuta, T., *et al.* (2001) Design and construction of a series of compact humanoid robots and development of biped walk control strategies. Robotics and Autonomous Systems, **37**(2–3), 81–100
4. Martinetz, T.M., Ritter, H.J., Schulten, K.J. (1990) Three-dimensional neural net for learning visuomotor coordination of a robot arm. IEEE Trans. on Neural Networks, **1**(1), 131–136
5. Jordan, M.I., Rumelhart, D.E. (1992) Forward models: Supervised learning with a distal teacher. Cognitive Science, **16**, 307–354
6. Xia, Y., Wang, J. (2001) A dual network for kinematic control of redundant robot manipulator. IEEE Trans. on Systems, Man, and Cybernetics – Part B: Cybernetics, **31**(1), 147–154
7. Kurematsu, Y., Maeda, T., Kitamura, S. (1993) Autonomous trajectory generation of a biped locomotive robot. IEEE Int. Conf. on Neural Networks, 1961–1966
8. Yamasaki, F., Miyashita, T., Matsui, T., Kitano, H. (2000) PINO, The humanoid that walk, Proc. of The First IEEE-RAS Int. Conf. on Humanoid Robots
9. Zhou, C., Meng, Q. (2003) Dynamic balance of a biped robot using fuzzy reinforcement learning agents. Fuzzy Sets and Systems, **134**(1), 169–187
10. Hirai, K. *et al.* (1998) The development of Honda humanoid robot. Proc. of the IEEE Int. Conf. on Robotics and Automation, 1321–1326
11. Espiau, B., Sardain, P. (2000) The antropomorphic biped robot BIP2000. Proc. of the IEEE Int. Conf. on Robotics and Automation, 3996–4001
12. Kaneko, K. *et al.* (2002) Design of prototype humanoid robotics platform for HRP. Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems, 2431–2436
13. De Lope, J., Zarraonandia, T., González-Careaga, R., Maravall, D. (2003) Solving the inverse kinematics in humanoid robots: A neural approach. Proc. of Int. Work Conf. on Artificial and Natural Neural Networks, IWANN'03 (to appear)
14. Lee, D.T. (1983) On finding the convex hull of a simple polygon. J. of Algorithms, **4**, 324–331

Neurosymbolic Integration: The Knowledge Level Approach

J. Mira¹, A.E. Delgado¹, and M.J. Taboada²

¹ Dpto. de Inteligencia Artificial
ETS Ing. Informática. UNED.
Madrid. SPAIN

{jmira, adelgado}@dia.uned.es

² Dpto. de Electrónica e Computación
Universidade de Santiago de Compostela. SPAIN
chus@dec.usc.es

Abstract. The time when the connectionist and symbolic perspectives of Artificial Intelligence (AI) competed against each other is now over. The rivalry was due essentially to ignorance on the implications of the knowledge level, as introduced by Newell and Marr. Now it is generally accepted that they are different and complementary forms of modeling and operationalizing the inferences in terms of which a problem solving method (*PSM*) decomposes a task. All these tasks, methods, inferences, and formal operators belong to a broad library of reusable components for knowledge modeling. The final configuration of a problem solving method, with symbolic and connectionist components, is only dependent on the particular balance between data and knowledge available for the specific application under consideration. Various approaches have been explored for neurosymbolic integration. In this paper we propose a classification of these approaches (unified, hybrid and system level) and strongly support that the integration has to be made at the knowledge level and in the domain of the external observer (the “house” of models).

1 Historical Perspective and Problem Statement

The purpose of knowledge engineering, the applied part of Artificial Intelligence (AI), is to build computable models of non-analytical human knowledge using symbolic, connectionist or hybrid *PSMs*. Analytical knowledge is dealt with by other branches of computation. Starting from the identification and preliminary analysis of the problem to be solved, the subsequent phases are: (1) knowledge modeling at the knowledge level and in the external observer domain (*EOD*), (2) operationalization of the inferences generated by the *PSM* used to decompose the task (still at the knowledge level, but now in the domain proper of the formal tools used), the own domain (*OD*), and (3) implementation of the formal counterpart of the model, passing from the knowledge level to the symbol level (the program), were both types of inferences end up being numeric.

In the development of these three phases there are two paradigms, which constitute two ways of modeling, operationalizing and programming the solution of a task [9]:

1. Symbolic computation, thick “grain” and programmable in a declarative manner.

2. Connectionist computation, small “grain” and partially “self-programmable” through learning. Here, a part of the knowledge is preprogrammed in the net structure itself. The rest of the knowledge is obtained by the net when trained with labeled data.

At the end, at the level of electronic processors, all computation is neither symbolic nor connectionist but numeric. Symbolism and connectionism are born in the domain of the external observer (the knowledge engineer) when nouns (concepts) and inferential verbs are associated to data structure and to the processes that handle them using precompiled semantic tables. That is to say, the difference between number (“labeled line”) and message appears in the domain of the human when we assign meaning to the entities of the input and output spaces, both being spaces of representation. Consequently, it is here, at the knowledge level and in the modeling and operationalization phases where the integration has to be made.

Knowledge modeling and operationalization started being connectionist with the pioneer works of W.S. McCulloch and W. Pitts [8], when in 1943 they introduced the first formal model, which we would today call minimal sequential circuit. This work constitutes the beginning of the connectionist approach to knowledge modeling and cognition. Much more recently this perspective has been integrated with the so called “situated cognition” [1] where more emphasis is put on the mechanisms underlying perception, decision-making and action control by means of feedback mechanisms than in the representational perspective of knowledge modeling. That is to say, in connectionism we look for nets of adaptive processors capable of solving problems (recognizing characters, inferring in a distributed manner) and learning by reinforcement and self-organization. In current terms we would say that this approach looks for *PSM* close to the physical (physiology or hardware) level, where structure and function coincide. It is at the source of distributed *AI* and it uses logic and analytics as the way to represent knowledge, a part of which is distributed in the net connections. The Hebb conjecture on the associative character of learning and the analogical models of Rosenblatt (perceptron and δ rule), Widrow-Hoff (Adalines and RMS rule), as well as some other proposals from Kohonen (associative memory and self-organizing feature maps) completed the panorama of the neurocybernetic stage of connectionist modeling [3].

Following this neurocybernetic precedent, the year 1956 is usually considered the year of the birth of the symbolic perspective of *AI* that in the first period (1956-1970) focused on problems characteristic of formal domains. In the mid-sixties, gradual changes took place in the symbolic perspective of *AI*, bringing them closer to the problems of real world and attributing increasing importance to the specific knowledge of the domain and, consequently, to the problems associated with its representation in terms of a “set of symbols” and subsequent use of these symbols in “reasoning”, without any reference to the neurophysiology and the mechanisms from which this knowledge emerges. This perspective gave rise to the “symbolic stage” (1970-1986) during which the emphasis was on technical tasks in narrow domains and programs to solve these tasks (“knowledge based systems” or “expert systems”). Knowledge modeling in the symbolic perspective is descriptive (facts plus rules) and different to the embodied character of connectionism.

Around 1986 there is a strong rebirth of connectionism, first as a competitive alternative to symbolic *AI* and then as a set of alternative or complementary methods

in the context of hybrid architectures, which combine multiple operators, both symbolic and connectionist.

The aim of this paper is to make some methodological considerations regarding the problem of neurosymbolic integration. In section two we review different strategies used for the integration. Then, in section three we expose the unified approach. Section four is dedicated to the cooperative architectures. In section five we present our knowledge level approach, where it is advisable to use both symbolic and connectionist operationalizations for different inferences working together. We conclude mentioning that practically all the inferences (“*select*”, “*compare*”, “*evaluate*”...) can be synthesized using either connectionist or symbolic operators and that the selected option is only dependent on the balance between knowledge and data available.

2 Different Strategies Used for the Neurosymbolic Integration

Symbolic methods of modeling and operationalization are proper for those situations in which we have all knowledge we need to solve a specific task, being the reasoning process of a deductive character. The static part of this knowledge is distributed into two layers: that one of the entities and relations of the application domain, and that other of tasks, methods, inferences, and roles which is independent of the domain.

On the contrary, connectionist methods of modeling and operationalization are proper for those other situations in which we have plenty of data (labeled or unlabeled) but little knowledge to solve the task. Consequently, we use the initially available knowledge for the configuration of the topology of the neural net (number of layers, number and type of local function for the “neurons” of each layer, learning algorithms and training procedure). The rest of the knowledge needed to solve the task has to be approximated by means of “accumulation of coincidences”, a process of inductive character. The reasoning process is then guided by data and we can also distinguish two layers: (1) The domain knowledge layer where we have the labeled lines to be used as input and output of the network and (2) the generic connectionist inferential scheme that links “observables” (inputs to the first layer) with “classes” (outputs of the neurons of the last layer). The connectionist *PSMs* enables us to reformulate any task as a multilayer classification task, being the inferences associated with the different layers of “neurons”.

The questions now are: (1) When is symbolic and connectionist methods integration needed?, and, (2) how must we do that integration?.

The answer to the first question is crystal clear; we need to integrate both types of methods in all those situations where we have neither all the knowledge nor all the data. And this happens to be in most of the cases in real world problems.

The answer to the second question is not so clear [2, 7] and several strategies have been proposed:

- *System level approaches* (use of ANN to obtain symbolic rules, neurofuzzy systems, adaptive resonance architectures -ART-...).
- *Unified approaches* (knowledge based inferential neural nets, connectionist expert systems).

- *Hybrid approaches* (hybrid “master-slave” architectures, knowledge level integration at the inferences operationalization phase).

In *system level approaches* the potential usefulness of both symbolic components for knowledge representation (rules, frames, ...) and inference as well as connectionist components for learning, adaptation and flexible modeling is well recognized. Consequently, ideas and operators from both fields are mixed together in a symbolic platform. Examples of this approach are the neurofuzzy systems, the ART architectures and the use of *ANN to obtain symbolic rules*. In this form of symbiosis, proposed by [6, 15, 16], we use the initial knowledge about a problem to define the architecture of the net, training the net later and re-transferring the content of the trained net to a symbolic representation (a set of rules).

3 The Unified Approach

This approach claims that we only need a unique sort of inferential component to cope with the needs of both symbolic and connectionist components. To make progress in this perspective we have at least two possible routes available to us and which can at first develop autonomously but must ultimately converge:

1. Extension of the computational capabilities of the connectionist model to include other forms of representation and inference, which are more powerful than the current ones (weighted sum followed by threshold). Connectionist mechanisms of learning still remain in this expanded version (supervised algorithms over a new set of parameters different from weights).
2. Reduction of the “grain” size of the symbolic model to access to the connectionist forms of representation (modular, distributed, small-grain and “self-programming” by learning).

The meeting point of these two routes is the use of parametric rules as models of local computation. This means to substitute the local function (weighted adder) and the threshold (sigmoid) by a conditional in the broad sense, as illustrated in figure 1. The antecedent of the rule evaluates the triggering conditions and the consequent selection of the proper decision. Both input and output roles are labeled lines (property-number). This extension eliminates the restrictions of the connectionist calculus (analog sum and non-linear decision function) and still retains the topology and learning facilities of neural nets. So we can say that we only need “inferential neurons” to cope with the representational and inferential power of “symbols” and with the learning facilities of “neurons”. A crucial point of this unified approach is the “physical character” of the connectivity between these inferential rules and, consequently, its computational limitations. In other words, not all the properties of symbolic systems using production rules (chaining by contents, induction, rule interpreter, agenda,...) can now be transferred to inferential neural nets which keep the topology and connectivity of conventional ANN (receptive fields, feedback, ...). Only those physically connected can be activated.

The connectionist *PSM* for the unified approach is of the type “*abstract-classify-refine*”, as shown in figure 2. The “*abstract*” inference includes all the data analysis, sensitivity studies and selection of labeled lines. Inference “*classify*” takes care of the numerical associations between the input and output lines and inference “*refine*” is in charge of the interpretation of the results, including the injection of semantics.

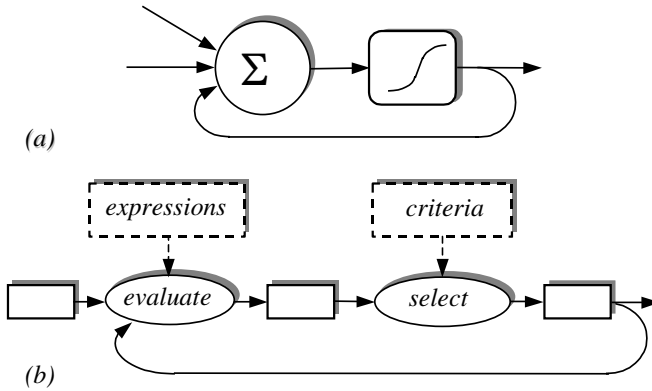


Fig. 1. Inferential “neurons”. (a) Analytical model. (b) Abstracted inferential scheme

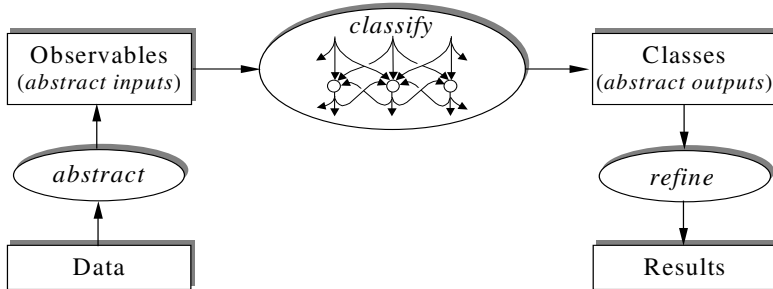


Fig. 2. A generic method for the unified approach

4 Hybrid Architectures

In this approach, symbolic and connectionist modules are interconnected using a hybrid architecture where the connectionist components (*C*) play the role of pre-processor, post-processor or sub-processor of a main symbolic processor (*S*), as illustrated in figure 3.

Although the description of *C* and *S* components is apparently peer, in fact it is the *S* component that takes the control, being *C* the subordinate component to which *S* accedes when needed. This vision of neurosymbolic integration is an implicit consequence of the usual strategy of implementing hybrid systems in terms of a computer program where the *C* components act as subroutines or functions called, when necessary, for the *S* components.

There is also the possibility of more complex hybrid architectures (“co-operative”) where the *S* and *C* components of a hybrid *PSM* cooperate in the solution of a task, sharing the domain ontology and the possibility for the entities (“concepts”) of the domain knowledge of playing input and output roles, both in symbolic and connectionist inferences. The only difference is then in the specific operators used in the formalization of each one of the *S* and *C* inferences (figure 4).

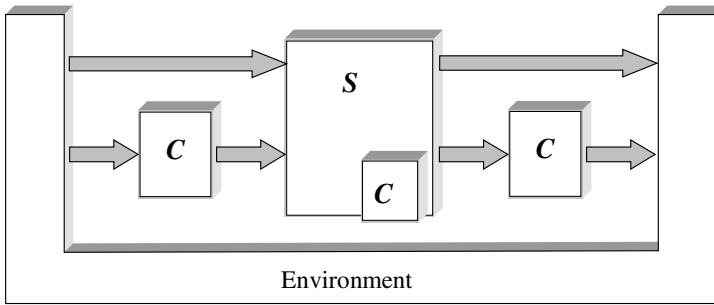


Fig. 3. Hybrid architecture in which the neural net acts as a pre-process, post-process or subordinated process of a main symbolic processor

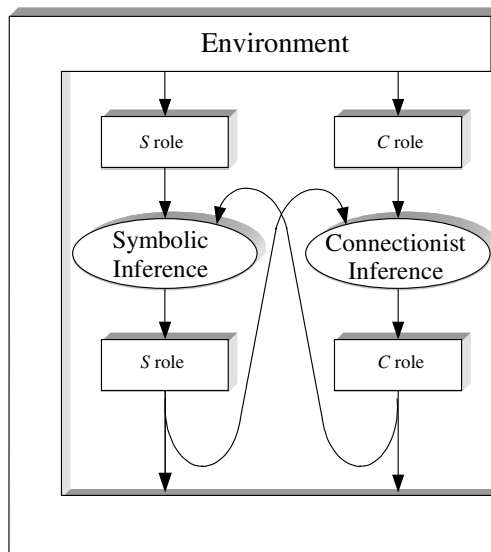


Fig. 4. A cooperative architecture for neurosymbolic integration. Input and output roles, and the underlying data structures, are compatible and usually shared

This integration strategy can again be classified in two groups: (1) simple cooperativity and (2) complex cooperativity, according to the method used to exchange information between the *C* and *S* modules.

In the *S-C* integration by means of simple cooperation, the task is decomposed in advance according to one or several *PSMs* and in such a way that each elementary inference has pre-assigned the *S* or *C* character before operationalization. Moreover, the function carried out by each one of these specific *S* or *C* modules is independent of the values of the inputs to other modules and also of the functions performed by these other modules. Finally, the integration of the different *S* and *C* inferences (the combination of its proposals) is made by means of an additional module, the “totalizer”.

This form of integration by simple cooperation corresponds to what in other taxonomies has been called “feeble integration” [4, 9, 14], although in these cases the

specification of the integration strategy is very close to the implementation phase. That is to say, the emphasis is put on the form of communicating and sharing the data structures, more than on the nature of the inferential scheme.

The integration by complex cooperation is closer to the concept of “complete” integration of [9] and to the concept of “strong integration” [4], defined in both cases in terms of shared data-structures. For us, the integration of S and C modules by complex cooperation is still an open problem, close to the unified approach with hybrid modules, both with S and C characteristics, activated in function of the influx data. Additionally, the functions of supervision and control have to be distributed. This leads S - C integration to the field of object oriented distributed AI. In this approach the computational capacity of the inferential neurons is still improved to become structured objects. In our terms, it would involve using frames to specify the neurons including fields of the following type [12]:

- S.1. Y_j^I is the *neuron-agent* which belongs to the l -th layer, where it holds position j .
- S.2. V_j^I is the *sampling area* where this neuron's inference field must go to look for data.
- S.3. V_j^{*I} is the *dialogue area* where the responses of other neurons which are going to participate in the inference of Y_j^I should be sought. It specifies the feedback of recurrent models and makes a certain type of backwards chaining possible.
- S.4. *Inference field*. It specifies the neuron-agent's local inference in the calculus mode. It includes as special cases the analogical and logical models. The step from analytics to general inference takes place when the linear or non-linear function of perceptrons, adalines or radial basis functions is substituted by the evaluation of a condition, and the threshold function by a conditional.

The condition section can include any combination of analogical or relational operators, but its complexity should be limited so as to admit “self-programming” by supervised or non supervised methods. Whatever operators are included in both fields, connectionist learning presupposes the existence of certain adjustable parameters which carry out the functions of the weights in the analytic models.

It is interesting to note that all the net's agents infer simultaneously, in a concurrent manner, and with intrinsic parallelism. In other words, there is no global inference which takes more time than the local inference of any neuron. These inferences take place at every synaptic delay. The neural net is thus interpreted as a structure of distributed micro-inferences. The function of each neuron-agent is to apply the knowledge of its field of inference to a local set of “true events” data in receptive field (in V_j and V_j^*).

- S.5. *Crossed models* {S.4 in $(y_k^q, \forall k \neq j \text{ and } q = 1, \dots, m)$ }. To be able to represent cooperative processes of a certain entity by means of neural nets, it is necessary for each neuron-agent to include an internal model of the agents with which it interacts. The most primitive version of the concept of model is

a copy of the field of inference (S.4) of the neighbors with which it dialogues V_j^* .

- S.6. *Field of learning.* Whatever the learning mechanism may be, the neuron-agent should have a meta-inference mechanism which allows it to modify the value of the parameters of the inference field (S.4), when it is in *training mode* and as a consequence of the selected criterion. Anything that may be modified should also include the *form* and *extension* of the receptive fields (V_j and V_j^*).
- S.7. *Mode control (M1, M2, M3).* For the neural net to accept an initial pre-programming and the alternative activity of calculation and learning, a mode control field is necessary. During mode *M1*, the architecture of the net is specified, and the most advisable initial values of the parameters are also given. For it, we use the initial available knowledge to solve the problem. Mode *M2* corresponds to inference from the current value of the parameters according to the field (S.4). Finally, mode *M3* corresponds to the execution of the learning algorithms (S.5 field) to modify the parameters of S.4 which will operate in the following interval of time.

5 The Knowledge Level Approach: Selection of S/C Options at the Operationalization Phase

Most of the problems of neurosymbolic integration are associated with the proximity of the decision to the implementation phase. If we consider the integration task from the beginning of the knowledge modeling phase, at the knowledge level [7, 13], and in the domain of the external observer [11], things become more clear, precise and unequivocal (figure 5).

We can decide for each task, *PSM*, subtask or inference the most adequate option (*S* or *C*), according to the balance between data and knowledge available at each specific phase of decomposition of the task, and for each particular domain of application. Awaiting new methodological developments to guide us in choosing the most adequate combination of *S* and *C* methods to resolve each concrete problem, we can abide by the following guidelines [12]:

1. First, the computational demands of a problem must be analyzed to find the task or combination of tasks, which best fits these demands. There are situations in which at this early stage we can already decide what the most adequate *PSM* to decompose the task is (hierarchical classification, multilayer perceptron with supervised learning, self-organizing maps, committee machines, fuzzy logic, neurofuzzy classifiers, and so on).
2. The most usual situation, however, is that we will need to continue the problem analysis with the decomposition of the task, first in subtasks and then in terms of a set of primitive inferences (“*establish*”, “*refine*”, “*select*”, “*match*”, “*evaluate*”, ...), and an inferential circuit connecting these inferences through dynamic roles. That is to say, at the level of task and first *PSM* (“*heuristic classification*”, for example) we could still have not clear the balance between knowledge and data.

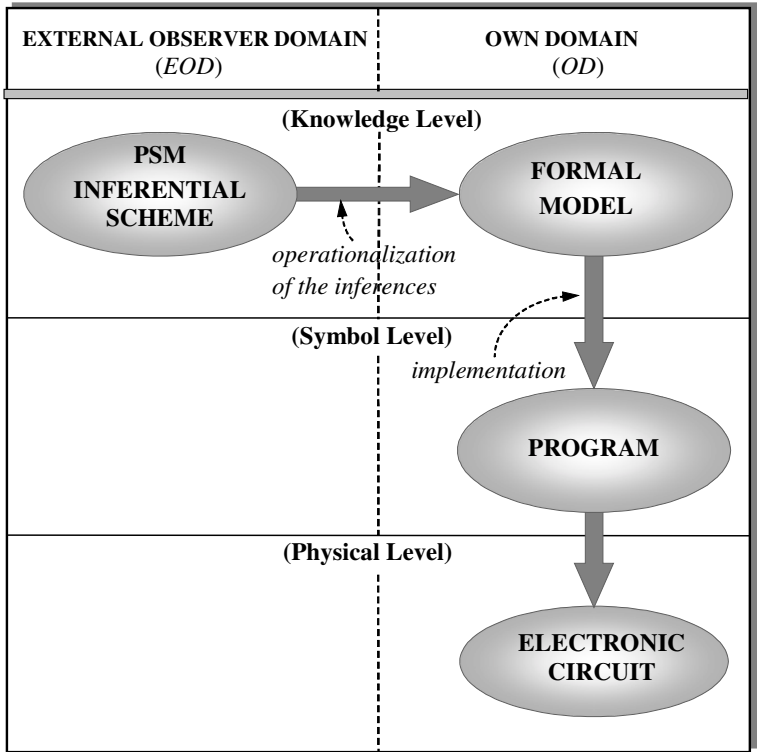


Fig. 5. The knowledge level approach. Integration is made at knowledge level and in the domain of the external observer, during the operationalization of the inferences

Nevertheless, in the next step (“*establish*” and “*refine*”), it could be clear that we have enough knowledge for “*establish*” but little knowledge for “*refine*”. Then, the inference “*establish*” is modeled using deterministic rules and for the “*refine*” inference we use neural nets, as illustrated in figure 6.

Then, *S* and *C* inferences must be described at the knowledge level and in the domain of the external observer, giving way to a diagram of functional interconnections according to the *S* or *C* model of each inference. If there are inferences for which it is possible and advisable to use neural nets, it is necessary to go down to the multi-layer model and to the learning mechanism to determine which is more adequate for each inference.

3. The next step is the operationalization of all the inferences (*S* and *C*), rewriting them in terms of formal entities and operators unequivocally understandable by a programming language.
4. Finally, we arrive to the implementation phase. Here we have to take care of making compatible the data structures and the roles (“observables”, “classes”, “measures”, ...) that participate in the interplay between *S* and *C* inferences.

There are few integration-oriented environments. The MIX platform proposes a remarkable solution to the implementation of *S-C* systems based on the encapsulation, in a multiagent architecture, of the *S* and *C* components in order to offer homogeneous interfaces [2].

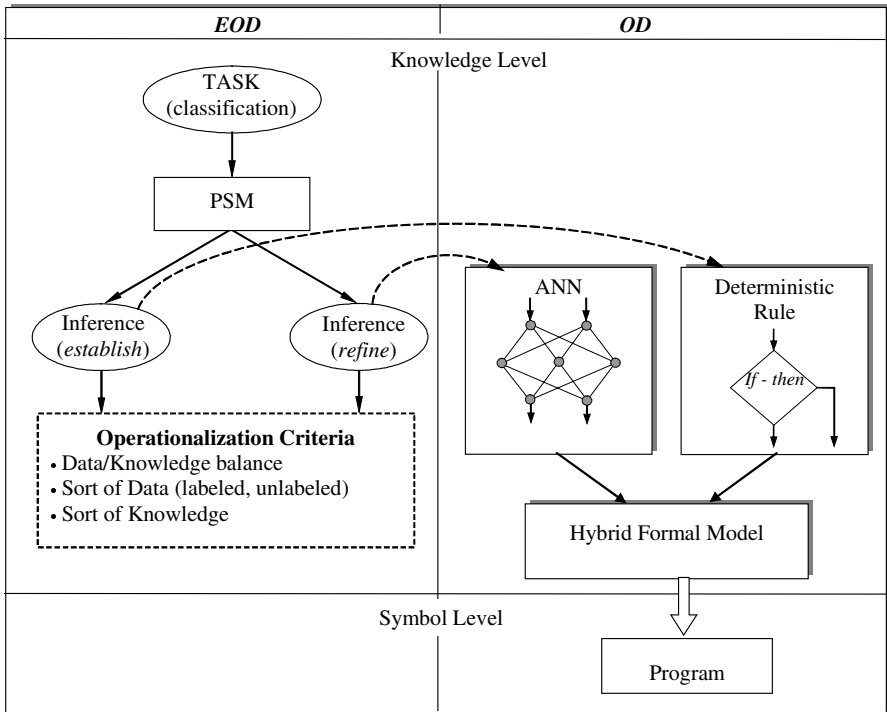


Fig. 6. Graphic summary of the knowledge level approach to neurosymbolic integration

6 Conclusions

In the previous sections of this paper we have examined a set of alternative approaches to neurosymbolic integration. The aim now is to make some final reflections regarding the similarities between symbols and numbers at the modeling phase. The differences between symbols and labeled lines (connectionist data) only appear in the *EOD* and at the knowledge level, before the operationalization phase. Thus each primitive inference has the possibility of being operationalized by means of a declarative mixture of data and deterministic rules or, alternatively, by means of neural nets. In the latter case, the data are labeled lines for the input (“observations”) and output (“classes”) of each layer of the net. With the labeling of these input and output lines in the *EOD*, the difference between “message passing” (symbolic modeling) and “number passing” (connectionist modeling) is blurred. In both cases, the meaning of “symbols” and “labels” remain at the boundary between *EOD* and *OD*. The same happens with the inferences.

Let us consider the generic inferential verb “*select*” defined as: “filtering one or more objects from a collection of several objects according to some selection criteria”. A symbolic formalization of this inference describes the set to which the selection process is applied as input role with data structure type list. The selected subset (output role) is also a list. Finally the selection criteria play the *static support role* and are usually formalized by means of a rule or set of rules.

A connectionist formalization of the “*select*” inference describes the same set to which the selection process is applied in terms of a set of numeric labeled lines. The output of the ANN that operationalizes this “*select*” inference is again a set of numeric labeled lines of the same cardinal. The active output lines reflect the subset selected. Finally, the static support role used as selection criteria is operationalized now by means of the supervised learning process and the labeled data used during the ANN training phase.

With the exception of the specific operators used in the final formal model (analogical, logical, gradient methods, weighted adders, look-up tables, and so on) there is nothing resident in the final program that could tell us the “*symbolic*” or “*connectionist*” character of the inferential model used as starting point of this program. A relevant part of the work to be done in neurosymbolic integration is concerned with the development of new libraries of reusable components for the operationalization of the “*usually symbolic*” inferences (neuronal “*compare*”, “*evaluate*”, “*abstract*”, “*refine*”, ...). And here is where our work is now going on.

References

1. Clancey, W.J.: *Situated Cognition*. Camb. University Press (1997)
2. González, J.C. Velasco, J.R. and Iglesias, C.A.: A distributed platform for symbolic-connectionist interoperation. In R. Sun and F. Alexandre (eds.) *Connectionist-Symbolic Integration*. 175–193. Lawrence Erlbaum Asoc. London (1997)
3. Haykin, S.: *Neural networks. A comprehensive foundation*. Prentice Hall (1999)
4. Hilario, M. et al.: Integration of model-based reasoning and neural processing. *Report of MIX Project*. CUI. University of Geneva (1999)
5. Hillario, M., Llamement, Y. and Alexandre, F.: Neurosymbolic integration: Unified versus hybrid approaches. *The European Symposium on Artificial Neural Networks*. Brussels, Belgium (1995)
6. Kuncicky, D.C., Hruska, S.I. and Lacher, R.C.: Hybrid systems: The equivalence of rule-based expert system and artificial neural network inference. *International Journal of Expert System*, 4(3): (1992) 281–297
7. Marr D.: *Vision*. Freeman, New York (1982)
8. McCulloch, W.S., and Pitts, W.: A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5 (1943) 115–133
9. Medsker, L.R.: *Hybrid Neural Network and Expert Systems*. Kluwer Academic Publishers, Boston (1994)
10. Mira, J. and Delgado, A.E.: Computación neuronal. En *Aspectos Básicos de la Inteligencia Artificial*, 485–576. Sanz y Torres. Madrid (1995)
11. Mira, J. and Delgado, A.E.: Where is knowledge in robotics? Some methodological issues on symbolic and connectionist perspectives of AI. In Ch. Zhou, D. Maravall and Da Rua (eds.) *Autonomous Robotic Systems*. Ch. I, 3–34. Physica-Verlag. Springer-Verlag. (in press)
12. Mira, J. et al.: *Aspectos básicos de la inteligencia artificial*, Sanz y Torres. Madrid (1995)
13. Newell, A.: The knowledge level. *AI Magazine* (summer 1981) 1–20
14. Sun, R. and Bookman, L.: How do symbols and networks fit together. *IA Magazine* (summer 1993) 20–23
15. Towell, G.C. Shavlik, J.W.: Refining symbolic knowledge using neural networks. In R. Michalshi., G. Tecuci (eds.) *Machine Learning*. Vol. IV:405–429. Morgan-Kaufmann Pub. C.A. (1994)
16. Towell, G.G.: Symbolic knowledge and neural networks: insertion refinement and extraction. *Technical Report 1072*, Univ. of Wisconsin-Madison, Computer Science Dept., January (1992)

On Parallel Channel Modeling of Retinal Processes

J.C. Rodríguez-Rodríguez¹, A. Quesada-Arencibia¹, R. Moreno-Díaz jr.¹,
and K.N. Leibovic²

¹Instituto de Ciencias y Tecnologías Cibernéticas
Universidad de las Palmas de Gran Canaria, Spain
{jcarlos,aquesada,rmorenoj}@ciber.ulpgc.es

²Department of Physiology and Biophysics
State University of New York at Buffalo, USA
bphknl@acsu.buffalo.edu

Abstract. Based on previous work on parallel processing for visual motion detection and analysis ([1], [2], [3]) in which interesting relationships among number and random distribution of cells, overlapping degree and receptive field size with fault tolerance, accuracy of computations and performance cost were shown in practice, we now focus our attention on modeling a two parallel channel visual peripheral system consisting of three layers of neuron-like cells that account for motion and shape information of perceived objects. A tetra-processor UltraSparc SUN computer is used for simulation and video-outputs in false color show the two-channel activity of the system. A train of input images presenting a moving target is analyzed by the neuron-like processing layers and the results presented as a video movie showing a colour coded version of neural activity.

1 Introduction

Parallel processing of signals, convergence and divergence of information lines and layered distribution of processors are ubiquitous characteristics of all known nervous systems and their perceptual peripherals. It is also an accepted working basis to see information processing in the first stages of the visual pathway in vertebrates as divided “horizontally” (or normal to the flowing direction of information) into layers, as the places where local operations like contrast detection take place, increasing the semantic content of the information as it goes from layer to layer, and “vertically” (or in the same direction of the information flow) into channels, where certain characteristics of the visual world are specifically treated like color, movement or shape of objects.

Based on this bio-inspired processing scheme, we have developed an artificial vision system in which we mimic the behaviour and characteristics found in natural systems. Particularly, we refer to systems where, basic units of process with complex connectivity, working in a parallel and distributed way, lead to a higher robust architecture from which an emergent behaviour arise.

The formal model used here is based on previous works and results obtained by Leibovic [4] and the biological studies of Barlow [5]. The basic ideas are the following:

- It has been demonstrated that it is possible to obtain measures of some geometric properties of the objects in the image generating a random group of receptive fields with computing capability in an artificial retina.
- It has been proposed a model in where the main factor is not the computation of the individual units but the information generated as a consequence of the individual outputs integration. Thus, a multilevel information processing system is obtained, where a first layer output feed a second layer and so on.
- It has been confirmed that a multichannel system is plausible in nature.
- It has been found and modeled a group of contrast detector cell and motion detector cells with a preferred direction. This second group of cells reacts positively when they detect movement in its preferred direction. In another case they inhibit its output.

2 Conceptual Model

2.1 General Description

Being models inspired in the actual description of nerve tissues, the system is composed by layers. Each layer executes a particular task. These computing layers include sets of artificial nerve cells-like units. The original image is processed by the layers according to a serial and parallel planning.

To start with, a set of general procedures is applied when the image reach our artificial retina. These procedures normalize the image for the subsequent high-level processing.

As shown in Fig. 1, in the following step we apply specific procedures for obtaining local conclusions. The information begins its diverging pathway thorough multiple channels at this point. These procedures can be independent among themselves. Thus, it is possible to perform several independent computations (N) to that common process. The channels execute their process in parallel respect to others in the same level.

After channel processing in which some specific characteristics of the image have been obtained, we would have integrative layers of information for obtaining global conclusions. This is the first convergent layer where the information is refined for increasing in complexity by iteration of this process. Information integration of the different channels is made in a high level representation, therefore the more you deepen into higher layers the less data you have and the more complexity you get, also increasing the semantic level of the neural description.

Coming back to our actual model, a first process to attenuate the image noise and the spurious signals is applied. Once this basic preprocessing is finished, channels take over the task: we apply procedures of detection of shape and motion in parallel, being both channels implemented in a completely independent way: no interchange of information is done between them. After the divergence stage has described the observed scene in terms of basic shape and movement data, there is a stage of convergence. We have thought up a new type of process unit that centralizes the information of the multiple channels. In this way we correlate the regions of the image with borders and vectors of movement which is equivalent to the description of the optic flow.

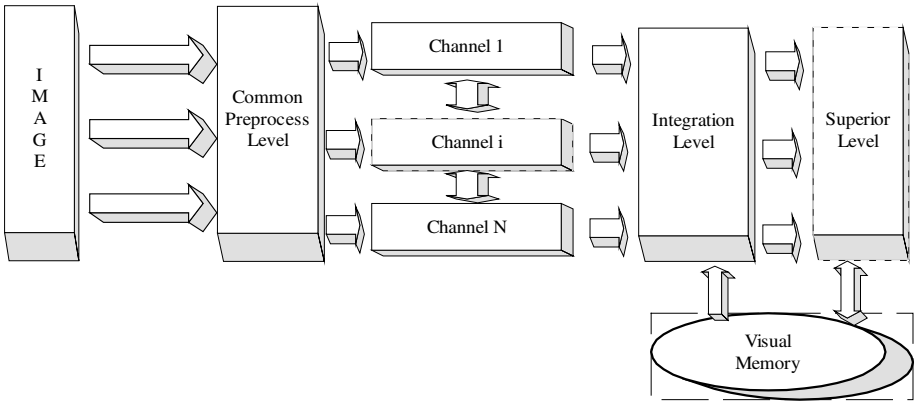


Fig. 1. General processing scheme

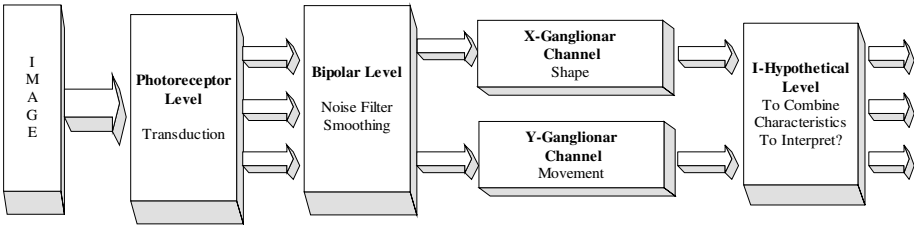


Fig. 2. Specific processing scheme applied to an image reaching our artificial retina

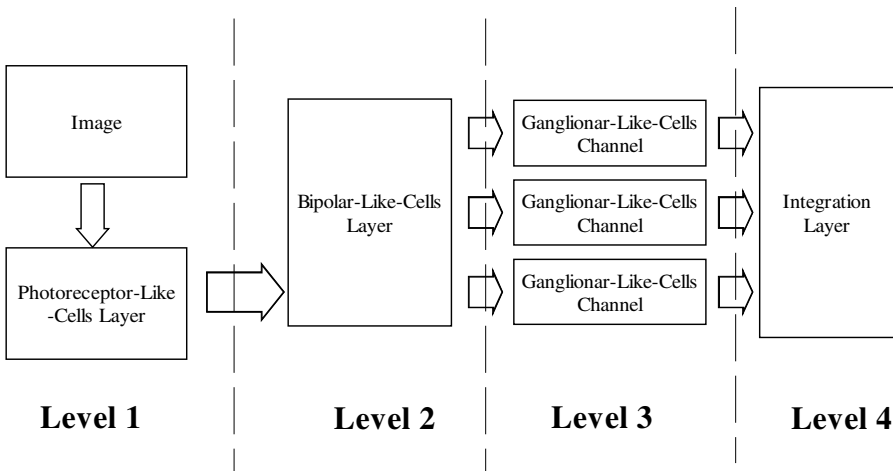


Fig. 3. In $t=1$ the first frame of the image is processed in level 1 (level 2, level 3 and level 4 are idle). In $t=2$ the first frame of the image is processed in level 2 (level 1, level 3 and level 4 are idle). Therefore only a level works at same time

2.2 The Natural Parallel Processing Counterpart

The basic scheme of the visual machinery upon which the explained system is inspired is shown in Fig. 3.

Information proceeds feed-forward, no feedback is considered, and it is assumed that a clock-like mechanism rules the flowing of information thorough the layers: the information doesn't proceed to the next layer until it has been completely processed in the previous one, thus providing a coherent spatio-temporal description of the outside world. This scheme, that is adopted only for practical implementing purposes, implies certain inefficiency of the computing blocks, that are unused part of the time, but avoids the mixing of temporally uncorrelated incoming data. Fig. 4 shows, as explained, the temporal development of tasks.

The implemented scheme is very similar to the second showed scheme except each layer is divided into four regions which are running in multitask mode.

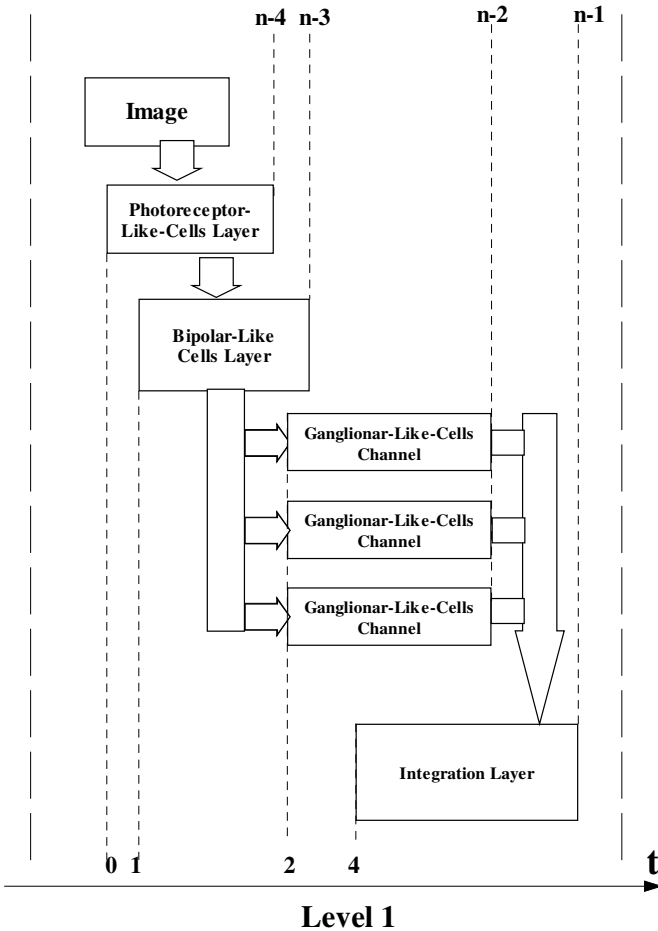


Fig. 4. In $t=1$ the first frame of the image is processed in level 1. In $t=2$ the first frame of the image is processed in level 2 and the second frame of the image is processed in leap 2, and so on

3 An Implementation of the Multichannel and Multilayer Proposed System

3.1 Receptive Fields

The receptive field is the data input field of the process unit. As in precedent works, receptive field overlap is present thru the system providing fault tolerance but also some spurious signals. This, together with the random distribution of cells, thus of receptive fields which are the counting units of the overall system, are the key characteristics of the model [1] [2]. As a result, we can easily vary the number of neurons and investigate it effects in the system performance [3].

The receptive fields have different shapes. The reason is that these receptive fields have been adjusted for being used with specific purposes (noise cleaning, transduction, motion or shape analysis). They can be punctual, circular without areas defined, circular with two zones (inner area and peripheral area). The punctual receptive field are used for transduction. The circular without zones are used for cleaning noise. The circular ones with two zones are used for the classic contrast detection.

There is a new kind of receptive fields: the angular fields which is thought to detect changes in one particular direction of movement. These receptive fields have two differentiated zones too. We have square receptive fields balanced out for integrating and obtaining macro conclusions.

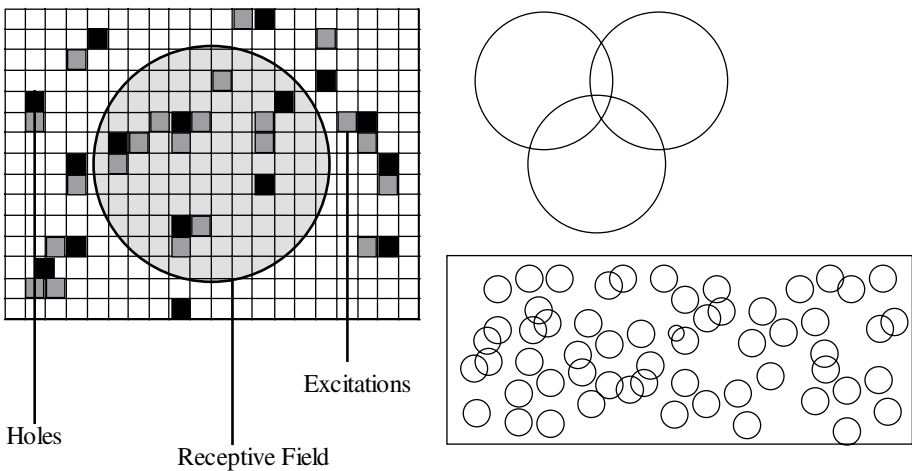


Fig. 5. Three properties of the receptive fields. Shape, overlap and distribution. The parameters are different in each layer

3.2 Transduction and Preprocessing Units

The artificial equivalent of photoreceptors and bipolar cells are in charge of pixeling the image and making some basic preprocessing as already explained. In the first

Photoreceptors cells

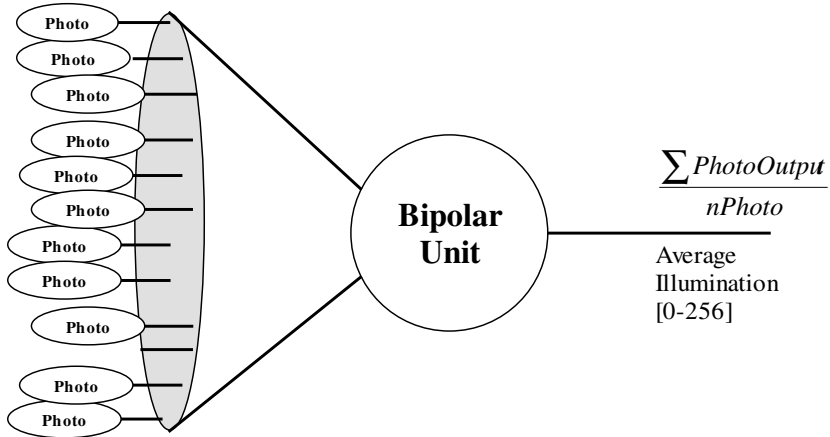


Fig. 6. The unit like bipolar-neuron. Circular Receptive Field. There is receptive field overlap. There are not holes. There are many photoreceptor-cells for every bipolar-cell

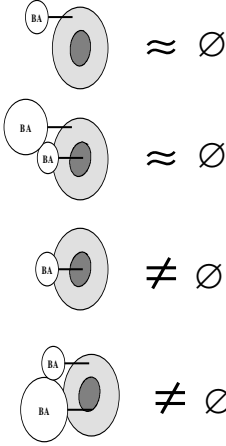
case, no overlapping is done since there is a one-to-one equivalence between pixels and transducing units, that compute in basis of luminance in the usual 0-255 gray scale. Bipolar-like cells clean the image by eliminating uncontrolled brilliances using a threshold mechanism which erases zones with high density of hushed light. Thus, a normalize image in terms of global measure of luminance is given after cleaning the noise due to the superfluous variations of luminosity. The process is shown in Fig. 6.

3.3 Channel Division: Ganglion Cell Processing of Shape and Motion

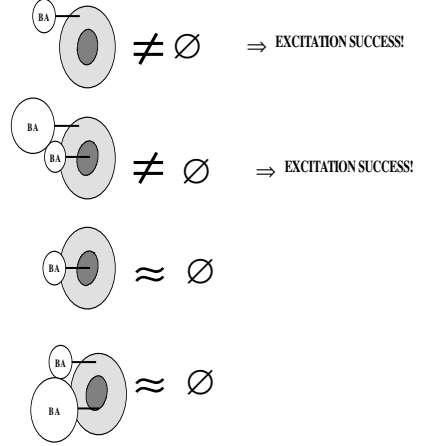
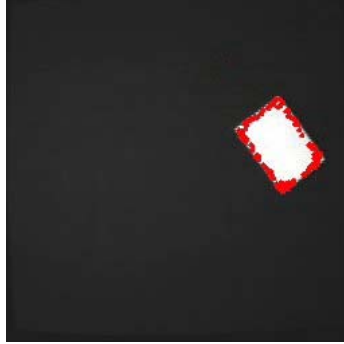
Once the image is homogenized in terms of regional luminance, the bipolar layer output flows thru separate channels, specialized in the description in terms of specific properties. The distributions of computing units of different channels, that correspond in our model to the X-Ganglion cells and Y-Ganglion cells neurophysiological pathways, are random but without coincidence, that is, receptive fields of X and Y-Ganglion cells computing units never cover the same area [6].

Shape Processing Using X-Ganglion Cells Channel. The X-ganglion units detect contrasts in the image. Any contrast is potentially interesting for the semantic, therefore this task is critical. The behavior of these units is classical: the mechanism uses center-periphery for the detection of contrasts. For example, an enough and positive difference of density of excited bipolars among the internal zone and the external zone implies cell activation. Again as described in the natural counterpart, On-center and Off-center cells are discriminated. The general performance of the channel is shown in Fig. 7 as a function of the density of bipolar cells activation. Fig. 8 shows the result of shape signaling of a moving squared object.

Behavior model X-Ganglionar Cell ON



Behavior model X-Ganglionar Cell OFF

**Fig. 7.** Behavior of X-ON-Ganglionar Unit**Fig. 8.** Excited like-X-ganglionar units react to contrast detection (as can be seen in the contour of the object)

$$F_{xON} = \frac{nIBA}{nIB} \times \left(1 - \frac{nPBA}{nPB} \right) \quad (1)$$

$$F_{xOFF} = \frac{nPBA}{nPB} \times \left(1 - \frac{nIBA}{nIB} \right) \quad (2)$$

where:

nIBA: number of excited Bipolar-neurons from inner circle.

nIB: number of total Bipolar-neurons from inner circle.

nPBA: number of excited Bipolar-neurons from periphery circle.

nPB: number of total Bipolar-neurons from periphery circle.

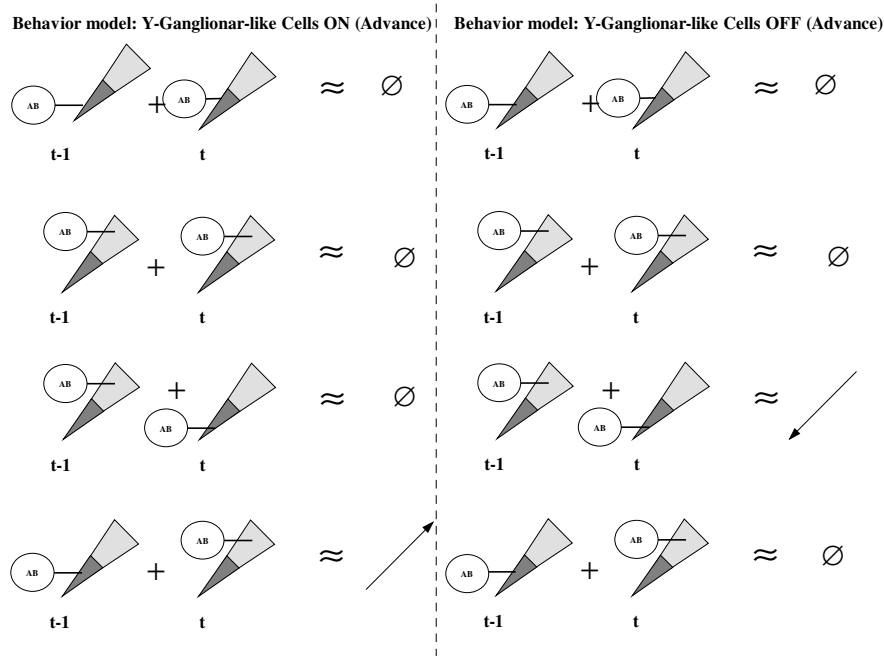


Fig. 9. Behavior of Y-Ganglionar Unit

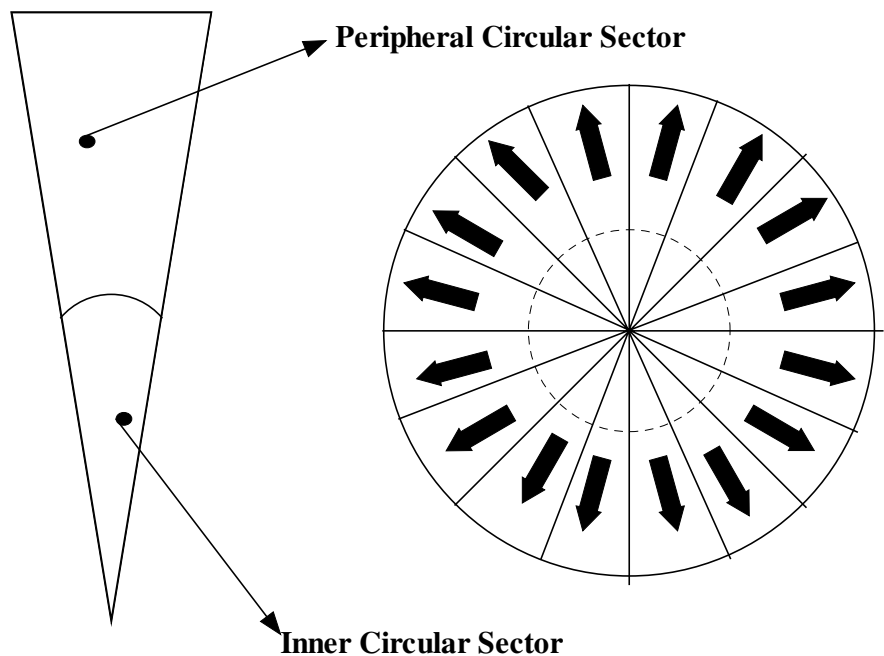


Fig. 10. Zones in the like-Y-Ganglionar unit: the inner circular sector and the periphery circular sector

Movement Processing by the Y-Ganglion Cells Channel. The Y-ganglion units are sensitive to the movement. Its receptive fields spread over frames separate in time but consecutive or very near. They compare the past and the present to determine when there are changes. These complex units are divided in categories specialized in direction selectivity. The implementation support sixteen different directions in circular sectors as shown schematically in Fig. 9.

In this case a change in the concentration of active bipolar cells in two consecutive frames excite the like-Y-ganglionar units. However, the changes of concentration must be originated from within to outside in the sensible direction. Fig. 10 show the different types of the like-Y-ganglionar units and the mathematical description of their activity is defined in (3), Y-Ganglionars-Like Cells ON-Advance, and (4), Y-Ganglionars-Like Cells OFF-Retreat. Fig. 11 shows the representation of Y-Ganglion cells activation after detecting the leading edge of a moving object, allowing for its subsequent segmentation.

$$\frac{nIBA(t-1)}{nIB} \gg \frac{nPBA(t-1)}{nPB} \Rightarrow F = \text{Max}\left(\frac{nPBA(t) - nPBA(t-1)}{nPB}, 0\right) \quad (3)$$

In – Another – Case $\Rightarrow F = 0$

$$\frac{nIBA(t-1)}{nIB} \ll \frac{nPBA(t-1)}{nPB} \Rightarrow F = \text{Max}\left(\frac{-nPBA(t) + nPBA(t-1)}{nPB}, 0\right) \quad (4)$$

In – Another – Case $\Rightarrow F = 0$

where:

nIBA: number of excited Bipolar-neurons from inner circular sector

nIB: number of total Bipolar-neurons from inner circular sector

nPBA: number of excited Bipolar-neurons from periphery circular sector

nPB: number of total Bipolar-neurons from periphery circular sector

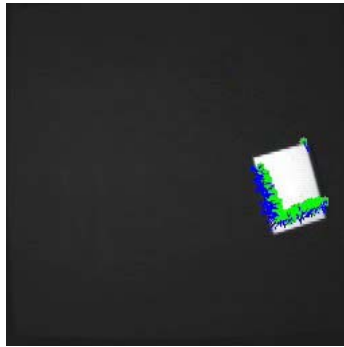


Fig. 11. We see the nucleus of the like-Y-ganglionar units that has become excited in gray color painted on the image. The straight lines indicate the direction that correspond to the excited like-Y-ganglionar unit. In this example we see the excitation of the advance like-Y-ganglionar unit. They react when the movement invades its receptive field

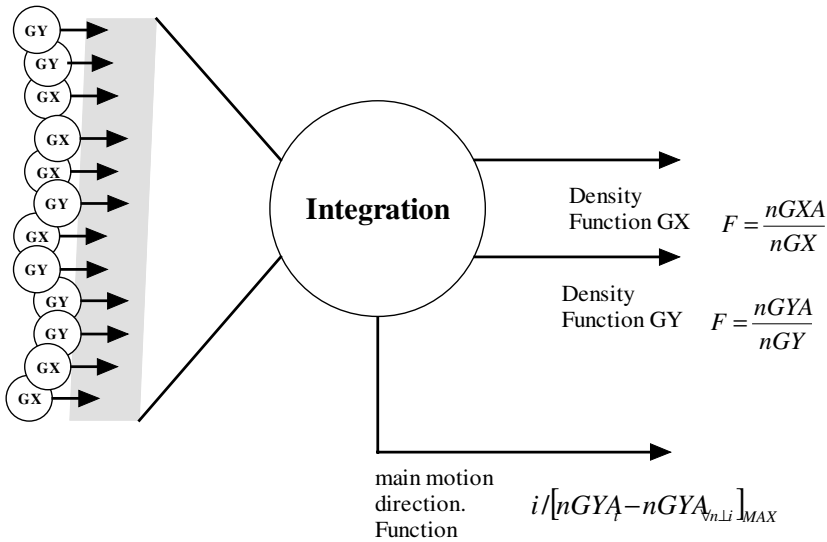


Fig. 12. A sketched of the proposed description integration unit

4 Conclusions and Future Work

A first attempt of channelled visual processing is presented, which includes several features that are described by the classical neurophysiology data of natural vision systems. Separate channels are used for shape and motion analysis of objects in real time, yielding coherent and synchronous descriptions of a moving object. Density of active cells is used as the basic data for shape computations, while motion discrimination is obtained and refined by the use of directional sensitive units, also described by neurophysiology. Separation of channels, and thus specialization of computing pathways, allow redundancy, fault tolerance and quick characteristic extraction from visual objects, but increases computational needs. A rationale for coping with the competing demands of coherent description and computational cost is, thus, pointed out. Also, integration in an unique description object is needed, being this the direction of part of our future work already sketched in Fig. 12.

References

1. Alemán-Flores, M., Leibovic, K.N., Moreno-Díaz jr, R.: A Computational Model for Visual Size, Location and Movement, Springer Lecture Notes in Computer Science, Vol. 1333. Springer-Verlag, Berlin Heidelberg New York (1997) 406–419
2. Quesada-Arencibia, A. Moreno-Díaz jr, R., Alemán-Flores, M., Leibovic, K.N.: Two Parallel Channel CAST Vision System for Motion Analysis. Springer Lecture Notes in Computer Science, Vol. 2178. Springer-Verlag, Berlin Heidelberg New York (2001) 316–327

3. Quesada-Arencibia, A.: Un Sistema Bioinspirado de Análisis y Seguimiento Visual de Movimiento. Doctoral Dissertation. PhD Thesys. Universidad de Las Palmas de Gran Canaria (2001)
4. Leibovic, K.N.: A Model for Information Processing with Reference to Vision. *Journal of Theore. Biology*, 11, (1996) 112–130
5. Barlow, H.B., Hill, R.M., Levick, W.R.: Retinal Ganglion Cells Responding Selectively to Direction and Speed of Image Motion in the Rabbit, *Journal Physiology*, 173, (1964) 377–407
6. Moreno-Díaz, R.: An Analytical Model of the Group II Ganglion Cell of the Frog's Retina. Report, Instrumentation Laboratory E1858, Massachusetts Institute of Technology, Cambridge, Mass, USA, (1965)
7. Moreno-Díaz jr, R., Leibovic, K.N., Moreno-Díaz, R.: Systems Optimization in Retinal Research. *Springer Lecture Notes in Computer Science*, Vol. 565. Springer-Verlag, Berlin Heidelberg New York (1992) 539–546

Geometric Image of Statistical Learning (Morphogenetic Neuron)

Elisa Alghisi Manganello and Germano Resconi

Department of Mathematics and Physics, Catholic University,
Via Trieste 17, 25100 Brescia, Italy
`elisa.am@libero.it`
`resconi@numerica.it`

Abstract. We present a neural model called *Morphogenetic Neuron*. This model generates a geometric image of the data given by a table of features. A point in the n -dimensional geometric space is a set of the values of the attributes of the features. In each point we compute a value of a field by a linear superposition of the values of the attributes in the point. The coefficients of the linear superposition are the same for all the points and are invariant for any symmetric transformations of the geometric space. The morphogenetic neuron can compute the coefficients by data without recursive methods, to reproduce the wanted function by samples (classification, learning and so on). Non-linear primitive functions cannot be represented in the morphogenetic geometric space. Primitive non-linear functions are considered as new coordinates for which the dimensions of the space are incremented. The geometry in general is non Euclidean and its structure is determined by the positions of the points in the space. The type of geometry is one of the main difference respects to the classical statistical learning and other neuron models. Connection between statistic properties and coefficients are founded.

1 Introduction

The machine learning, statistical learning and support vector machine have witnessed a resurgence of interest over the last few years. Data is not longer a scarce resource, it is abundant and its analysis needs "intelligent" methods to extract relevant information from it. We need to find a general rule that explains data given only one set of samples of limited size, as requested in the learning process. If the objects under consideration are associated with target values, the learning problem is called supervised; otherwise, if the data is a sample of objects without target values, the learning problem is called unsupervised. Statistical learning takes care of risk and conditional probability in the supervised and unsupervised learning. One of the devices mostly used to describe the learning process is the Support Vector Machine theory, where a kernel operator and features define an Euclidean vectorial space, in which data are located and separated, according to their class values. Moreover, we know that current theory of multisensory representation is inconsistent with the existence of a large proportion of multimodal

Table 1. Samples of values for the attributes.

	Basis functions or attributes			
Objects	ψ_1	ψ_2	$\dots$	ψ_q
Ob_1	$\psi_1(Ob_1)$	$\psi_2(Ob_1)$	$\dots$	$\psi_q(Ob_1)$
Ob_2	$\psi_1(Ob_2)$	$\psi_2(Ob_2)$	$\dots$	$\psi_q(Ob_2)$
$\vdots$	$\dots$	$\dots$	$\dots$	$\dots$
Ob_n	$\psi_1(Ob_n)$	$\psi_2(Ob_n)$	$\dots$	$\psi_q(Ob_n)$

neurons with gain fields and partially shifting receptive fields and do not fully resolve the recoding problem and statistical issues involved in the multisensory integration. As published by Lynn C. Robertson different areas of the cortex receive sensory information through different receptors, corresponding to different features of the sensory mosaic. This modular organization of the brain led to the question of how features that are registered separately are bound together to produce our unified experience of the world (binding problem). We argue that the local approach suggested by recent theories based on an Euclidean space is the main reason of loss of efficiency in classical methods. We propose an alternative theory, based on a neural architecture that combines basis functions with the superposition. We use basis function to solve the recoding problem and superposition for statistical inference, This new architecture is called Morphogenetic Neuron.

2 Description of the Context by Reference of the Feature

Given a context with a set of basis features (attributes) $\psi_1, \psi_2, \dots, \psi_q$, which values are represented in Table 1.

The distinction between objects and features allows us to define two different spaces: the space of the objects, which is Euclidean, and the space of features, which is a subspace of the space of objects and whose coordinates are the attributes themselves, seen as vectors of the space of the objects. Actually, the feature space can be non Euclidean. Each vector of the space of features is seen as a linear superposition of the attributes.

When we assume that the features are the coordinates of our space that define our context, every vector in this space can write as a superposition of the coordinates or feature in this way

$$\mathbf{v}_k = \sum_{j=1}^q w^j X_{j,k}, \quad (2.1)$$

where $X_{j,k} = \psi_j(Ob_k)$ and w^j (for all $j = 1, 2, \dots, q$) are the contravariants components of the vector $\mathbf{v}_k$. The vector $\mathbf{v}_k$ is a vector in the subspace of the space of the objects included in the space generated by the vectors of the feature space. We point out that the space is not obtained by a formal and abstract

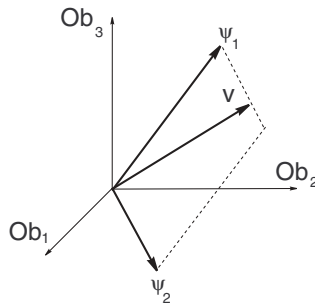


Fig. 1. Space of objects and it subspace of features

Table 2. Samples of values for two colors and three objects

	Red	Green
Table	1	0
Chair	0	1
Lamp	1	1

definition, but is consequent of special piece of data that are considered as basis data, by which we can create the best model for all the other data. In this way, data are considered as an *inseparable unity*, which the best image is the non-Euclidean space.

(2.1) Example. Let’s consider as an example of the space of features the space of colors defined in table 2. A vector **v** can be written in the following way:

$$\mathbf{v} = w^1(1, 0, 1) + w^2(0, 1, 1) \tag{2.2}$$

If $w^1 = 2$ and $w^2 = 0,5$ we have

$$\mathbf{v} = 2(1, 0, 1) + 0.5(0, 1, 1) = (2, 0.5, 0.5). \tag{2.3}$$

We can give a graphic representation, as shown in Fig. 2.

The space of features defines a subspace of the objects where all possible vectors **v** are located, obtained by a weighted superposition of the vectors of the space of features. The weights of the superposition are the contravariant components of the vector **v**.

The following figures (Fig. 3 and 4) show us the distinction between contravariant and covariant coordinates in a two-dimensional non Euclidean space:

The covariant coordinates are defined as:

$$w_j = \sum_{k=1}^n \mathbf{v}^k X_{j,k} \tag{2.4}$$

that represents the discrete scalar product between the vector **v** and the features.

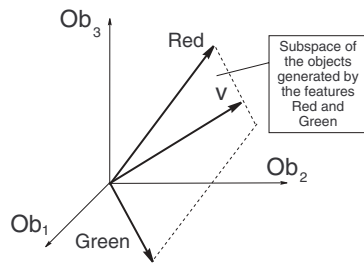


Fig. 2. Space of objects and features Red and Green

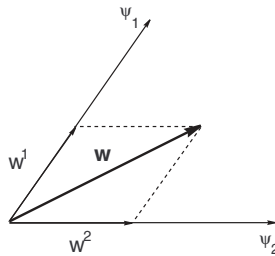


Fig. 3. Controvariant coordinates w^j .

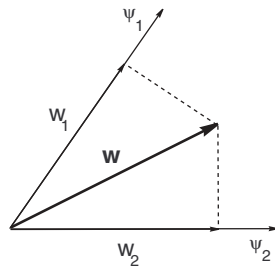


Fig. 4. Covariant coordinates w_j .

If the space is Euclidean (see Fig. 5), the controvariant and covariant coordinates coincide:

If we substitute eq(2.4) into eq(2.1) we obtain:

$$w_j = \sum_{i=1}^n \sum_{k=1}^n w^i X_{i,k} X_{j,k} = \sum_{i=1}^n w^i g_{i,j}, \quad (2.5)$$

where

$$g_{i,j} = \sum_{k=1}^n X_{i,k} X_{j,k}. \quad (2.6)$$

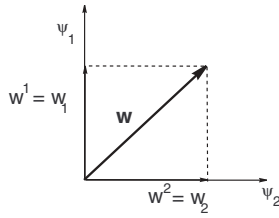


Fig. 5. Covariant and contravariant coordinates.

So we have that

$$g = \begin{bmatrix} \sum_{i=1}^n X_{1,i}^2 & \sum_{i=1}^n X_{1,i}X_{2,i} & \cdots & \sum_{i=1}^n X_{1,i}X_{q,i} \\ \sum_{i=1}^n X_{1,i}X_{2,i} & \sum_{i=1}^n X_{2,i}^2 & \cdots & \sum_{i=1}^n X_{2,i}X_{q,i} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^n X_{1,i}X_{q,i} & \sum_{i=1}^n X_{2,i}X_{q,i} & \cdots & \sum_{i=1}^n X_{q,i}^2 \end{bmatrix}, \quad (2.7)$$

and

$$\begin{bmatrix} w^1 \\ w^2 \\ \vdots \\ w^q \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n X_{1,i}^2 & \sum_{i=1}^n X_{1,i}X_{2,i} & \cdots & \sum_{i=1}^n X_{1,i}X_{q,i} \\ \sum_{i=1}^n X_{1,i}X_{2,i} & \sum_{i=1}^n X_{2,i}^2 & \cdots & \sum_{i=1}^n X_{2,i}X_{q,i} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^n X_{1,i}X_{q,i} & \sum_{i=1}^n X_{2,i}X_{q,i} & \cdots & \sum_{i=1}^n X_{q,i}^2 \end{bmatrix}^{-1} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_q \end{bmatrix}. \quad (2.8)$$

The tensor g is called *fundamental tensor* and it allows us to pass from covariant coordinates to contravariant ones. But not only: the fundamental tensor describes all the intrinsic properties of the space it represents. In its matrix expression (eq.(2.7)) the diagonal terms are the self-correlation of the features and the diagonal terms represent the mutual correlation among the features.

(2.8) Remark. The fundamental tensor describes the distance between two points in general non Euclidean coordinates, given by the following expression:

$$S = \sum_{i=1}^n g_{i,j} w^i w^j. \quad (2.9)$$

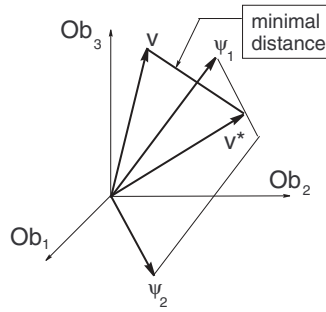


Fig. 6. The vector $\mathbf{v}$ and its approximation $\mathbf{v}^*$.

If the space is Euclidean, the fundamental tensor is the identical tensor and the distance S is the ordinary distance:

$$S = \sum_{i=1}^n (w^i)^2. \quad (2.10)$$

(2.10) Remark. If in table (1) we have that $n > q$, it is possible that a vector $\mathbf{v}$ is not inside the subspace of the feature. When we project the vector $\mathbf{v}$ on the vectors of the feature we obtain the covariant components of the vector $\mathbf{v}$ as we can see in eq(2.5). With the tensor g we can obtain the contravariant components from the covariant. In conclusion we can compute a vector $\mathbf{v}^*$ by the contravariant components. Because $\mathbf{v}$ is outside the space generated by the feature space, we have that $\mathbf{v}$ is different from $\mathbf{v}^*$. So, we have built the vector $\mathbf{v}^*$, that is an approximation of $\mathbf{v}$, and its distance from the external vector which distance from the external vector $\mathbf{v}$ is the minimal (See Fig. 6). In Fig. 6 we show the distance between the external vector $\mathbf{v}$ and the internal vector $\mathbf{v}^*$ to the space of the feature. When $\mathbf{v}^*$ is calculated by the tensor g , the distance between $\mathbf{v}$ and $\mathbf{v}^*$ is the minimal distance among all possible distances between $\mathbf{v}$ and one vector inside the subspace of the features.

3 A Complete Example of the Two Dimensional Space of Features

Given two features (X, Y) , n objects and one external vector of the objects F , we can write the following table, according to the general formula given in Table 1.

Table 3. Two features and the vector depending on the n objects.

	X	Y	F
Ob_1	X_1	Y_1	$F(Ob_1)$
Ob_2	X_2	Y_2	$F(Ob_2)$
$\vdots$	$\dots$	$\dots$	$\dots$
Ob_n	X_n	Y_n	$F(Ob_n)$

For the previous Table 3 and for the definition of fundamental tensor g (2.6), we have:

$$g = \begin{bmatrix} \sum_{i=1}^n X_i^2 & \sum_{i=1}^n X_i Y_i \\ \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n Y_i^2 \end{bmatrix}. \quad (3.11)$$

The contravariant components are:

$$w_1 = \sum_{i=1}^n X_i F(Ob_i) \quad \text{and} \quad w_2 = \sum_{i=1}^n Y_i F(Ob_i), \quad (3.12)$$

that are given by the scalar products of the external vector F with the features vectors, that define the space. So the covariant components are (according to eq.(2.8)):

$$\begin{bmatrix} w^1 \\ w^2 \end{bmatrix} = g^{-1} \begin{bmatrix} \sum_{i=1}^n X_i F(Ob_i) \\ \sum_{i=1}^n Y_i F(Ob_i) \end{bmatrix}. \quad (3.13)$$

where

$$g^{-1} = \frac{1}{\sum_{i=1}^n X_i^2 \sum_{i=1}^n Y_i^2 - \left(\sum_{i=1}^n X_i Y_i \right)^2} \begin{bmatrix} \sum_{i=1}^n Y_i^2 & - \sum_{i=1}^n X_i Y_i \\ - \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n X_i^2 \end{bmatrix} \quad (3.14)$$

(3.14) Proposition. When $F(Ob_i) = \alpha X_i + \beta Y_i$ and $\sum_{i=1}^n X_i = 0$, $\sum_{i=1}^n Y_i = 0$ we have that:

$$w^1 = \alpha \quad \text{and} \quad w^2 = \beta. \quad (3.15)$$

According to equation (2.1) we have

$$V_k = w_X^1 k + w^2 Y_k, \quad (3.16)$$

and consequently

$$V_k = F(Ob_i) = \alpha X_i + \beta Y_i \quad (3.17)$$

(3.17) Proposition. Even with a rotation or a translation of the coordinates X and Y , the coordinates w^1 and w^2 do not change. For $X' = AX$ and $Y' = BY$, the coordinates w^1 and w^2 become:

$$w^1 = A\alpha \quad \text{and} \quad w^2 = B\beta. \quad (3.18)$$

The contro-variant coordinates w^1 and w^2 are invariant for translation and rotation. For a change in the scale of the space (X, Y) , the coordinates change in a proportional way to the change of the scale.

(3.18) Proposition. Given a function $F(Ob)$ different from the samples of coordinates X and Y , we can firstly compute the form $V_k = w^1 X + w^2 Y$ and then determine the difference:

$$F(Ob) - V_k = \Delta F(Ob). \quad (3.19)$$

For the function $\Delta F(Ob)$ we can repeat the same process and we obtain a new linear form

$$\Delta V_k = \Delta w^1 X + \Delta w^2 Y; \quad (3.20)$$

we can go on repeating the same process until $w^1 = 0$ and $w^2 = 0$. The last function obtained will be impossible to write as a linear form of X and Y . In this way we can divide the function $F(Ob)$ in two parts

$$F(Ob) = w^1 X + w^2 Y + G(Ob). \quad (3.21)$$

where $G(Ob)$ cannot be represented in a linear form. Because not all the functions $F(Ob)$ can be written in the linear form $F(Ob_i) = \alpha X_i + \beta Y_i$, we can change the dimension of the space by including samples of functions not linear. In this case we introduce a non linear basic function $G(Ob)$.

(3.21) Example. Given the Boolean function $X \equiv Y$ (equivalence between the variables X and Y), it would be possible to prove that does not exist a set of coordinates w^1 and w^2 for which

$$[X \equiv Y] = w^1 X + w^2 Y. \quad (3.22)$$

The function $X \equiv Y$ is a non linear function. When including this non linear function, the space of the features becomes $S = (X, Y, X \equiv Y)$.

We could prove that all the Boolean functions, except for the tautology and the absurd functions, can be written as a linear combination of the previous three functions:

$$F(X, Y) = w^1 X + w^2 Y + w^3 (X \equiv Y). \quad (3.23)$$

For instance, the implication function $F = X \rightarrow Y = \neg X \vee Y$, when $X, Y \in -1, 1$ can be written as

$$F = -X + Y + (X \equiv Y). \quad (3.24)$$

Even if talking about Boolean functions of Boolean variables, we have considered the values for the variables symmetrical, giving value 1 for True, -1 for False attributions. When X is different from Y (for instance X is true and Y is false) we have $F = -1 - 1 - 1 = -3 < 0$, so the value of F will be considered false, and so on, it is possible to reconstruct the values of the implication functions.

When X is true and Y is true too, $X \equiv Y$ is true and gives a positive increment to the function F . When X is different from Y , $X \equiv Y$ gives a negative effect on F .

(3.24) Remark. Boolean functions of two variables are fourteen (without counting tautology and absurd) and, as we have seen, the basis function are only three, joined in a linear form. In this case we obtain a reduction of the complexity in the representation of the Boolean functions. Possible extension to a generic set of points in the space (X, Y) are possible.

4 Statistical Properties of the Controvariant Coordinates

According to eq.(3.17) and to eq.(3.13), we can write:

$$\begin{aligned} V_k = A & \left[\left(\sum_{i=1}^n Y_i^2 \right) X_k - \left(\sum_{i=1}^n X_i Y_i \right) Y_k \right] + \\ & + B \left[\left(\sum_{i=1}^n X_i^2 \right) Y_k - \left(\sum_{i=1}^n X_i Y_i \right) X_k \right], \end{aligned} \quad (4.25)$$

where

$$A = \frac{\sum_{i=1}^n X_i F(Ob_i)}{\sum_{i=1}^n X_i^2 \sum_{i=1}^n Y_i^2 - \left(\sum_{i=1}^n X_i Y_i \right)^2}, \quad (4.26)$$

and

$$B = \frac{\sum_{i=1}^n Y_i F(Ob_i)}{\sum_{i=1}^n X_i^2 \sum_{i=1}^n Y_i^2 - \left(\sum_{i=1}^n X_i Y_i \right)^2}. \quad (4.27)$$

When $\sum_{i=1}^n X_i = 0$ and $\sum_{i=1}^n Y_i = 0$, according to J.P.Guilford ([15], page 91),

$$r_{Y,X}^2 = \frac{\left(\sum_{i=1}^n X_i Y_i \right)^2}{\sum_{i=1}^n X_i^2 \sum_{i=1}^n Y_i^2}, \quad (4.28)$$

is the correlation index.

When $r_{Y,X}^2 = 1$, we have that

$$\sum_{i=1}^n X_i^2 \sum_{i=1}^n Y_i^2 - \left(\sum_{i=1}^n X_i Y_i \right)^2 = 0, \quad (4.29)$$

A and B assume a singular value. When

$$\left(\sum_{i=1}^n X_i^2 \right) Y_k - \left(\sum_{i=1}^n X_i Y_i \right) X_k = 0 \quad (4.30)$$

or

$$X_k = \frac{\sum_{i=1}^n X_i Y_i}{\sum_{i=1}^n Y_i^2} Y_k, \quad (4.31)$$

that is the regression equation ([15], page 371), we have

$$V_k = B \left[\left(\sum_{i=1}^n X_i^2 \right) Y_k - \left(\sum_{i=1}^n X_i Y_i \right) X_k \right], \quad (4.32)$$

where

$$\left(\sum_{i=1}^n X_i^2 \right) Y_k - \left(\sum_{i=1}^n X_i Y_i \right) X_k = 0 \quad (4.33)$$

or

$$Y_k = \frac{\sum_{i=1}^n X_i Y_i}{\sum_{i=1}^n X_i^2} X_k, \quad (4.34)$$

is the second regression equation. In conclusion, the vector V_k in eq.(4.25) is the linear combination of the two regression equations

$$\left(\sum_{i=1}^n Y_i^2 \right) X_k - \left(\sum_{i=1}^n X_i Y_i \right) Y_k = C_1 \quad (4.35)$$

and

$$\left(\sum_{i=1}^n X_i^2 \right) Y_k - \left(\sum_{i=1}^n X_i Y_i \right) X_k = C_2 \quad (4.36)$$

(4.36) Remark. When the function $F(Ob) = k$, with $k \in \mathbb{R}$, and $\sum_{i=1}^n X_i = 0$

and $\sum_{i=1}^n Y_i = 0$, we have that $w^1 = 0$ and $w^2 = 0$

5 Conclusion

We introduced a novel generalization of the concept of neural unit, which has been named *Morphogenetic Neuron*. From an abstract point of view, it is a generic analog input-output device through which two elementary operations are possible: the "Write" and "Read" operation, i.e.:

1. the operation *Write*: starting from a suitable reference basis functions and a set of samples of the function F it generates the non Euclidean space and compute the controvariant coordinates w^k ;
2. the operation *Read*: starting from controvariant coordinates, compute the linear form $\mathbf{v}$ in eq.(2.1).

When the linear form is different from the function F , we can compute the difference between F and $\mathbf{v}$. On the difference we can repeat the steps as described in proposition (3). Finally, we can obtain a non linear function G . In this case from the dimension N , we move to a dimension $N + 1$, with the introduction of a new basic function G . We can repeat this process until we generate a suitable set of basic functions that give a complete image of the possible rules or functions that we want to use. Future work will try to give concrete applications of the Morphogenetic Neuron.

References

1. McCullough, Pitts W.H.: A logical calculus of the ideas immanent in nervous activity. [Bull. Math. Biophys. 5, 115–133, 1943.]
2. Robertson, L. C.: Binding, Spatial Attention and Perceptual Awareness. [Nature Review, Feb 2003]
3. Nobuo S.: The extent to which Biosonar information is represented in the Bat Auditory Cortex. [Dynamic Aspects of Neocortical Function, G.M. Edelman, John Wiley, New York 1984].
4. Resconi G.: The morphogenetic Neuron in Computational Intelligence: Soft Computing and Fuzzy. [Neuro Integration with Application, Springer NATO ASI Series F Computer and System Science; Okyay Kaynak, Lotfi Zadeh , Burhan Turksen , Imre J.Rudas; vol 162, pp. 304–331, 1998].
5. Resconi G., Pessa E., Poluzzi R.: The Morphogenetic Neuron. [Proceedings fourteenth European meeting on cybernetics and systems research, pag.628–633 April 14–17, 1998].
6. Murre J. M. J.: Learning and categorization in modular neural networks. [Erlbaum, Hillsdale, NJ, 1992].
7. Benjafield J. G.: Cognition.[Prentice-Hall, Englewood Cliffs, NJ, 1992].
8. Salinas E., Abbott L.F.: A model of multiplicative neural responses in parietal cortex. [Proc. Natl. Acad. Sci. USA Vol.93, 11956–11961, 1996].
9. Duif A. M., van der Wal A. J.: Enhanced pattern recognition performance of the Hopfield neural network by orthogonalization of the learning patterns. [Proc. 5th Internat. Parallel Processing symposium, Newport (CA), 10 pages, 1991].
10. Salinas E., Abbott L.F.: Invariant Visual responses From Attentional gain Fields. [The American Physiological Society, 3267–3272, 1997].
11. Resconi G., van der Wal A. J., Ruan D.: Speed-up of the MC method by using a physical model of the Dempster-Shafer theory. [Int. J. of Intelligent Systems, Special issue on FLINS'96, Vol. 13, Numbers 2/3, pp 221–242, 1998].
12. Resconi G., van der Wal A. J.: Morphogenetic neural networks encode abstract rules by data. [Information Sciences 142 (2002) 249–273]
13. Pouger A., Snyder L. H.: Computational approaches to sensorimotor transformations. [Nature neuroscience – supplement – volume 3 – November 2000, pag 1192–1198]
14. Riesenhuber M., Poggio T.: Models of object recognition, transformations. [Nature neuroscience – supplement – volume 3 – November 2000 pag.1199–1204].
15. Guilford J. P.: Fundamental Statistics in Psychology and Education. [International Student Education, New York McGraw-Hill, 1965]

Systems and Computational Tools for Neuronal Retinal Models

Roberto Moreno-Díaz and Gabriel de Blasio

Instituto Universitario de Ciencias y Tecnologías Cibernéticas
Universidad de Las Palmas de Gran Canaria, Spain
`rmoreno@ciber.ulpgc.es`
`gdeblasio@dis.ulpgc.es`

Abstract. This paper presents some systems theoretical and computational tools which allow for qualitative and quantitative models to explain the rate of firing (outputs) of ganglion retinal cells to various stimuli, mostly in higher vertebrates. Section 1 presents the neurophysiological bases of the formal tools, in which the main point is that specialized computation by some ganglion cells is to be performed at the inner plexiform layer, via amacrine cells. Section 2, presents the minima prerequisites for a qualitative model of linear and non-linear behavior of ganglia for higher vertebrates. The points here are that signals reaching the inner plexiform layer are fast and retarded versions of the input image, and that non-linear local lateral interaction accounts for non-linearities. Section 3, is devoted to computational representations that will permit to go from qualitative to quantitative formal models. The computational tools are based on generalized Newton Filters.

1 Introduction. Neurophysiological Considerations

The Retinal Receptive Field (*RRF*) of a ganglion cell is that part of the retina to which the ganglion cell is sensitive to. Ganglion cells responses to point-light or gratings stimuli are used to characterize *RRF* of cells and they have been classified according the types of responses. A large variety of sizes, shapes, and temporal behavior has been encountered for vertebrates from fish to mammalian, the description being typically performed in terms of excitatory and inhibitory zones [1].

Ganglion cells in vertebrate retinae can be classified in two limiting groups: a) Specialized complex ganglia, which are particularly selective to some spatio-temporal properties of stimuli, and b) non specialized simple ganglia. There is a range of intermediate types. The rule seems to be that specialized retinal ganglion cells are a majority in vertebrate having none or little developed visual cortex as frog and pigeon, whereas simple non specialized cells predominate in retinae such as cat's monkey's and probably man's, vertebrate with well developed visual cortex.

In cat's retina, simple cells (X cells and some Y cells) are in a large proportion [2]. Many of the simple cells present a range of quasilinear spatio-temporal input-output relation, with ON-centre. OFF-periphery types of responses or viceversa [3]. Their behavior can be, reasonably well modelled by linear spatio-temporal models. Marked non linear and more specialized cells may present non concentric receptive fields, local contrast detection and high sensitivity to the motion of stimuli in a preferred direction.

For cat's, and probably for other simple retinæ, complex and simple ganglion cells coexist in the same zones of the retina, although their densities differ from centre to periphery. This should discard the possible hypothesis that horizontal and bipolar cells are, in general, engaged in specialized computation, since each ganglion of a type seems not to have its own and exclusive set of bipolars. It may be concluded that complex computation is performed at the inner plexiform layer and/or at the ganglion cell body.

The inner plexiform layer is a logical site for specialized extraction of local properties of retinal stimuli, since spatial information is still present there prior to further integration by ganglia, and the layer is richly horizontally interconnected. In addition, amacrine and their synapses to ganglia seem to be related to complex operation. As the anatomy of the inner plexiform layer evidences, the density of synapses in complex retinæ is much larger than in simple ones; simple ganglia are mainly driven by direct contact with bipolars, whereas complex ganglia show a large number of amacrine synapses.

Various relevant accounts on the responses of avian retinal ganglion cells [4] indicate that their peculiarities include: a) insensitivity to the motion of any stimulus in the excitatory receptive field (ERF), with local either ON or OFF ERF's, and inhibitory surround. b) Low or high sensitivity to the motion of stimuli in all directions within the ERF, with local ON-OFF responses. c) Directional sensitivity to the motion of contrasts irrespective of their polarity, with local ON-OFF ERF's.

Retinal ganglion cells of a frog present marked specialized behavior [5]. Their functional complexity seems to be correlated with the complexity in shape and distribution of their dendritic trees. Thus, the inner plexiform layer seems to be the logical site where prominent specialization occurs through the interaction of bipolar axons, amacrine and ganglia dendrites.

When compared to simple retinæ, such as a cat's, a frog's ganglion cells show generalized strong non-linearities besides their apparent pattern recognition capabilities, so that responses are in many instances little dependent on stimuli parameters such as contrasts. This might be an indication that a local extraction of properties already happens at the outer retinal layers to a degree higher than in other vertebrate with well developed visual system. A systems approach to the frog's outer retinal layers non linear behavior, which is coherent with the systems tools presented here are presently under study.

2 Systems Tools

2.1 Non Linear Processing

From the systems theory point of view, the inner plexiform layer seems to be a definitive site for specialization. The probable operation there is non-linear lateral interaction in which amacrine cells play an important role. At least qualitatively, specialized properties can be the result of a lateral linear inhibition, plus local half-wave rectification.

For higher vertebrates input signals to the inner plexiform layer can be reduced to local (point) fast and retarded signals, approximately linearly related to the input signals. That is, a fast signal, $f(\mathbf{r}, t)$ and a retarded signal $f_R(\mathbf{r}, t)$. Let $f(\mathbf{r}, t)$ and $f_R(\mathbf{r}, t)$ be their Laplace transforms. The resulting two possibilities of lateral inhibition are:

$$X'_1(\mathbf{r}, s) = f(\mathbf{r}, s) - (k_R \cdot A_R) \int_{\mathbf{r}'} U(\mathbf{r}, \mathbf{r}') f_R(\mathbf{r}', s) d\mathbf{r}' \quad (1)$$

$$X'_2(\mathbf{r}, s) = f_R(\mathbf{r}, s) - (k/A) \int_{\mathbf{r}'} U(\mathbf{r}, \mathbf{r}') f(\mathbf{r}', s) d\mathbf{r}' \quad (2)$$

A_R and A are the areas of the corresponding inhibitory surfaces S_R and S ; k_R and k are the weighting factors.

Signals arriving to ganglion cells could be assumed to undergo processes similar to those in cells having quasilinear behavior: that is, they are affected by weighting factors and summated thereafter. From this, it follows that the output of a ganglion cell model at the origin is a pulse train of instantaneous frequency given by

$$G(t) = Pos \left[G_0 + \iint_{\mathbf{r}t'} K_1(\mathbf{r}, t - t') Pos X'_1(\mathbf{r}, t') d\mathbf{r} dt' + \iint_{\mathbf{r}t'} K_2(\mathbf{r}, t - t') Pos X'_2(\mathbf{r}, t') d\mathbf{r} dt' \right] \quad (3)$$

Where G_0 is the spontaneous response, when it exists; $K_1(\mathbf{r}, s)$ and $K_2(\mathbf{r}, s)$, Laplace transforms of $K_1(\mathbf{r}, t)$ and $K_2(\mathbf{r}, t)$, are the corresponding ganglion weights. For simplicity, we shall sometimes assume that K_1 and K_2 are independent of s .

Prior to the discussion of (3), we need to consider the operations leading to $X'_1(\mathbf{r}, t)$ and $X'_2(\mathbf{r}, t)$, according to (1) and (2). Linear models of simple ganglia suggest that $f(\mathbf{r}, s)$ and $f_R(\mathbf{r}, s)$, under the simplest temporal assumptions, may be related by

$$f_R(\mathbf{r}, s) = [1/(1 + \tau s)] f(\mathbf{r}, s) \quad (4)$$

$f(\mathbf{r}, s)$ may in turn be considered the result of a linear spatio-temporal transformation on some non-linear local function of the intensity of the incident light

at the retina. We shall assume that, for simple stimuli such as points, disc or bars with two levels of gray, $f(\mathbf{r}, t)$ essentially conserves the form of the stimuli, with little changes such as smoothing of discontinuities and spatial spreading. That is, the linear transformation leading to $f(\mathbf{r}, t)$ acts essentially as a low pass spatio-temporal filter, but not too low.

Equations (4) and (1) indicate that $X_1(\mathbf{r}, s)$ is the result of a linear transformation on $f(\mathbf{r}, s)$ by the factorized kernel

$$W(\mathbf{r}, \mathbf{r}', s) = \delta(\mathbf{r} - \mathbf{r}') - k_R U(\mathbf{r}, \mathbf{r}')/A_R(1 + \tau s) \quad (5)$$

For stationary input stimuli, (5) reduces to

$$W(\mathbf{r}, \mathbf{r}') = \delta(\mathbf{r} - \mathbf{r}') - k_R U(\mathbf{r}, \mathbf{r}')/A_R \quad (6)$$

When $\delta(\mathbf{r} - \mathbf{r}')$ is within S_R , (6) is a typical contrast detector kernel. For S_R a circle centered at r , the kernel is symmetric. Excitation cancels inhibition for $k_R = 1$, and there is no output for constant uniform stimulation. For $0 < k_R < 2$, there is stationary contrast detection, whereas for $k_R \geq 2$, a contrast could only be detected if it moves, since the inhibition from S_R is retarded.

When the surface S_R is not centered at r , such that $\delta(\mathbf{r} - \mathbf{r}')$ might be at the edge or even outside S_R , (5) is an asymmetric kernel. Asymmetric kernels with retarded inhibition provide for a maximum sensitivity to moving positive stimuli in a preferred direction, which goes from the site of the contribution of the fast signal to that of the retarded signal.

The time response in (5) can be found by stimulating with a temporal step of afferent stimuli covering a spot of a size about S_R at the origin. That is, $f(\mathbf{r}, s) = U(0, \mathbf{r})/s$. $\delta(\mathbf{r} - \mathbf{r}')$ is assumed to be within or at the edge of S_R . From (1)

$$sX'_1(\mathbf{r}, s) = U(0, \mathbf{r}) - k_R \varphi(\mathbf{r})/A_R(1 + \tau s) \quad (7)$$

$\varphi(\mathbf{r}) \equiv \int_{\mathbf{r}'} U(\mathbf{r}, \mathbf{r}')U(0, \mathbf{r}')d\mathbf{r}'$ is the area of the overlap between the inhibitory surfaces, $S_R(0)$ and $S_R(\mathbf{r})$, corresponding to positions $\mathbf{r} = 0$ and $\mathbf{r} = \mathbf{r}$, respectively. $X'_1(\mathbf{r}, t)$ is negative or zero for $\mathbf{r}$ outside $S_R(0)$.

For $\mathbf{r}$ within $S_R(0)$, (7) becomes

$$sX'_1(\mathbf{r}, s) = 1 - k_R \varphi(\mathbf{r})/A_R(1 + \tau s) \quad (8)$$

Equation (8) corresponds to a typical ON- process; there is, or not, sustained ON response depending upon the value of $k_R \varphi(\mathbf{r})/A_R$. At the origin, $\varphi(0) = A_R$, and for $k \geq 1$, there is not sustained response. If the kernel is symmetric and $k_R = 2$, there is no point with sustained response. Note that only positive values of $X'_1(\mathbf{r}, t)$ are considered in the discussion. Also, asymmetry does not destroy the uniform ON character of the afferent field, since there is a convolution, as indicated by (1). In sum, the kernel in (5) provides for local contrast detection, ON response and, if asymmetric, for higher sensitivity to moving positive

stimuli in the preferred direction. These properties are more or less emphasized according to the value of k_R .

Equations (2) and 4 show that $X'_2(\mathbf{r}, s)$ is also the result of a linear transformation on $f_R(\mathbf{r}, s)$ by the kernel

$$W(\mathbf{r}, \mathbf{r}', s) = [\delta(\mathbf{r} - \mathbf{r}')/(1 + \tau s)] - kU(\mathbf{r}, \mathbf{r}')/A \quad (9)$$

Equation (9) reduces to (6) for stationary stimuli. Arguments as before show that the kernel in (9) provides for contrast detection, OFF response and, if asymmetric, for higher sensitivity to moving decrements of stimuli in the preferred direction. All these properties depend upon the value of k . Simplest local ON and OFF response by kernels (5) and (9) may be obtained by assuming that the inhibitory surfaces S_R and S are very small, i.e., reduce to a point:

$$U(\mathbf{r}, \mathbf{r}')/A_R \rightarrow \delta(\mathbf{r} - \mathbf{r}')$$

$$U(\mathbf{r}, \mathbf{r}')/A \rightarrow \delta(\mathbf{r} - \mathbf{r}')$$

Local contrast detection and possible sensitivity to the direction of motion of stimuli are then absent.

2.2 Linear Processing

For cuasi-linear ganglion cells, the output of the ganglion cell model given by (3) should be compatible with linear operation for some simple cells; this must correspond to the case of negligible lateral interaction. From (1) and (2), negligible lateral interaction implies $k_R = k = 0$. Also, the spontaneous response is null, $G_0 = 0$. Equation (3) reduces to

$$G(t) = Pos \left[\iint_{\mathbf{r}t'} K_1(\mathbf{r}, t - t') Pos f(\mathbf{r}, t') d\mathbf{r} dt' + \iint_{\mathbf{r}t'} K_2(\mathbf{r}, t - t') Pos f_R(\mathbf{r}, t') d\mathbf{r} dt' \right] \quad (10)$$

If the additional requirement is made that $f(r, t)$ and $f_R(r, t)$ must be positive (10) leads to

$$G(t) = Pos L^{-1} \left[\int_{\mathbf{r}} K_1(\mathbf{r}, s) Pos f(\mathbf{r}, s) + K_2(\mathbf{r}, s) Pos f_R(\mathbf{r}, s) d\mathbf{r} \right] \quad (11)$$

Equation (11) corresponds to a general expression for linear invariant spatio-temporal models under the "two paths" hypothesis. Central mechanism corresponds to kernel $K_1(\mathbf{r}, s)$ and peripheral mechanism to kernel $K_2(\mathbf{r}, s)$. For ON-centre, OFF-periphery. K_1 , is excitatory (positive) and K_2 , is inhibitory (negative). A more elaborated relation between $f_R(\mathbf{r}, s)$ and $f(\mathbf{r}, s)$ could be

postulated, although (4) provides for the main features relating a signal and its retarded.

The spatial shapes of kernels K_1 and K_2 are a matter of arguments among physiologists and vision theoreticians. There is a very elegant proposal of solution coming from systems computation theory as it will be considered in the next section.

3 Computational Tools

3.1 Computational Structures for Lateral Interaction

As it has been pointed out in the previous sections, a structure widely accepted, (for lateral interaction at different retinal layers) is that of a more or less concentric center-surround input fields. Signals from these zones interact in general non-linearly in time and space, so that if I_{ijk} is the intensity of the stimulus on photoreceptor (i, j) at time k , the instantaneous frequency of firing of the cell at the origin $(0, 0)$ and time t is $F(I_{ijk})$ for $(i, j) \in RRF$ and $k = (t-1, t-2, \dots, t-\tau)$, where τ is the 'memory' of the cell and of the cells impinging on it [6].

We first focus on some aspects of the approximately linear behavior, where F is assumed to be a linear function. The assumption of almost linearity leads to an output from the layer of the type

$$F(I_{ijk}) = \sum_{i,j,k} \alpha_{ijk} I_{ijk} \quad (12)$$

Time invariance (no learning or adaptation), implies that α_{ijk} is of differences on k , that is $\alpha_{ijk} = \alpha_{ij}(t-k)$. From this, it follows that an appropriate model for the cell spatio-temporal behavior, must include an 'input space' formed by temporal 'layers' of spatial inputs I_{ijk} for k from $(t-1)$ to $(t-\tau)$, where the time convolution type of integral given by F is performed. For cat's X cells it has been argued (ref) that $\alpha_{ij}(t-k)$ can be approximately be represented by two sets of space-time weights corresponding to center and periphery. The most popular sets of weights [7], correspond to the so called 'Gaussian center-surround model', which corresponds to $\alpha_{ijk} = \alpha(t-k)e^{-a(i^2+j^2)} - \beta(t-k)e^{-b(i^2+j^2)}$, α and β are usually taken as 'fast' and 'retarded' lag kernels, or viceversa.

In what respects to space (time abstracted), and alternative to the 'difference of Gaussians' approach, from the optimum filter for contrast (edge) detection [8] consisted in the so called Laplacian of a Gaussian. The Laplacian of a Gaussian in fact corresponds to the second of the Hermite functions (changed in sign), which functions are in turn, the generalization to the continuum of the so called Newton Filters [9].

The theory of Newton Filters provides, in fact, for a very useful paradigm of retinal computation of the quasi-linear behavior and a very transparent way to relate function to structure and to introduce 'credible' non-linearities to model Y cells and other in lower vertebrates, including the extra disinhibitory or inhibitory surrounds [10],[11].

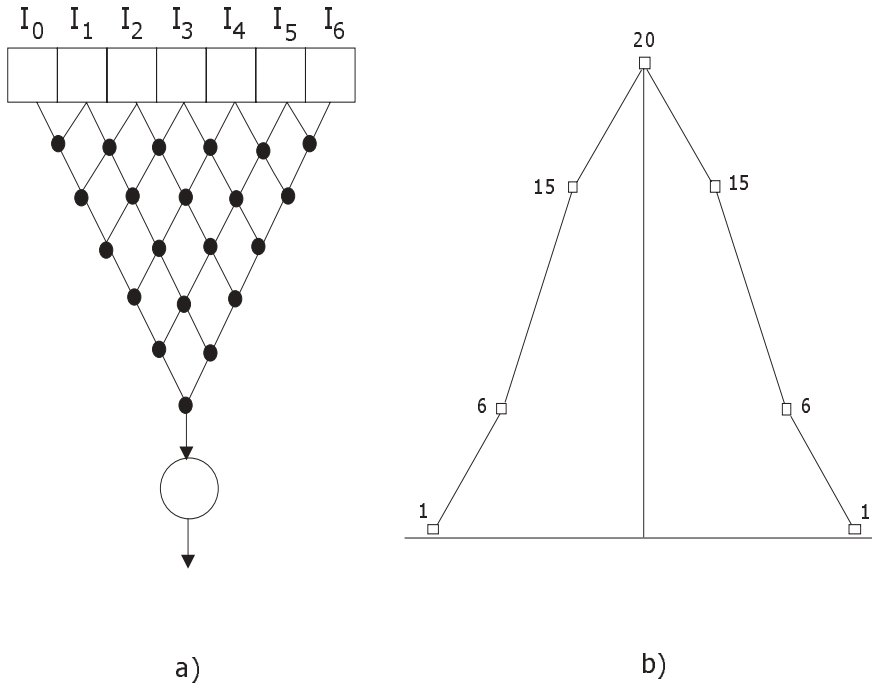


Fig. 1. a) Newton Filter structure for seven receptors, $N(A7,D0)$. Local weights are all +1. b) Weight representation for $N(A7,D0)$

Consider the one dimensional structure of figure 1 which can be regarded as a cell C computing on a linear "retina" of 7 receptors which provide inputs $I_0, I_1, \dots, I_6$ through a kind of "dendritic tree". The operation on each node of the dendritic tree is first assumed to be simply the addition of the two signals coming down to each node. In this case, we have what is called a "Newton Filter" type A (purely additive), of length $6+1$. The name "Newton Filter" is because the output C is a weighted addition (linear operation) on I_i , that is,

$$C = \sum_{i=0}^6 k_i I_i \quad ; \quad k_i = \binom{6}{i}$$

k_i are the coefficients of the Newton binomial. In general, Newton Filters may consist of additive or subtractive rows. They are denoted $N(A_m, D_n)$, where m is the number of rows with addition, and n is the number of subtracting (difference) rows. It is irrelevant what rows are additive or subtractive, but not how many there are of each type. The "length" of the receptive field is $L = m + n + 1$. The case of figure 1 corresponds to $N(A_7, D_0)$.

Newton Filters are a type of discrete linear filters. Their origin, mathematical properties and their extensions (Generalized Newton Filters, Causal Filters, Newton Transforms, Newton Convolutions, Global Transforms, Coding,

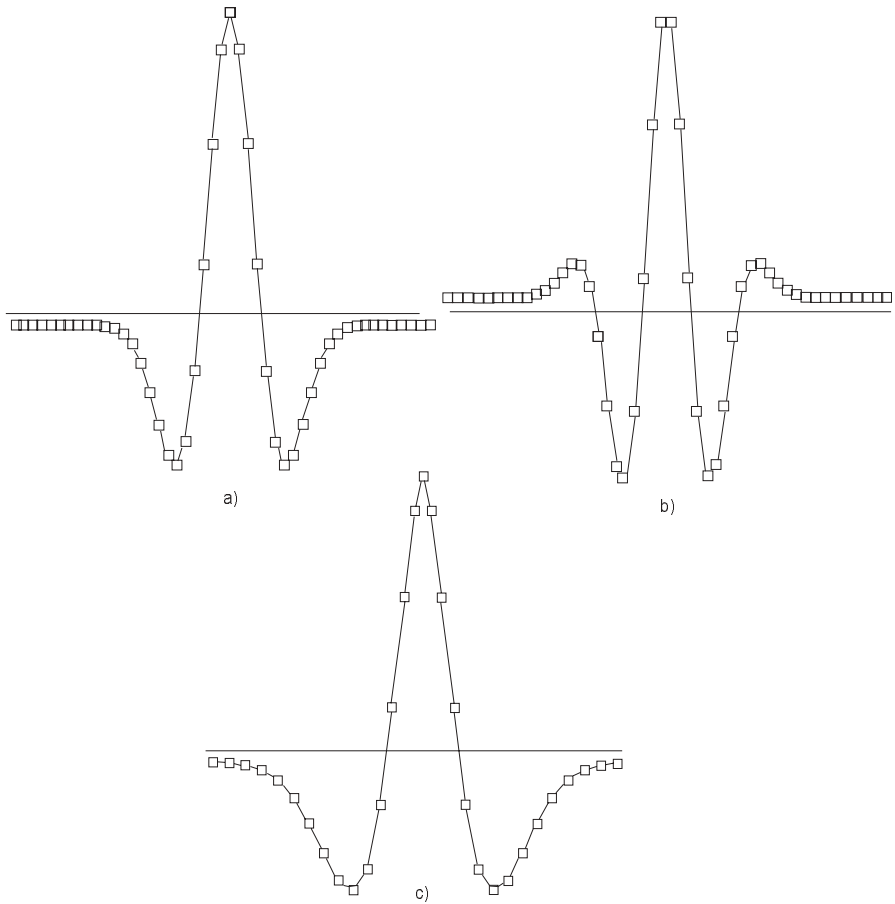


Fig. 2. a) Filter $-N(A49,D2)$. Observe the typical 'Mexican hat' profile. b) Filter $N(47,D4)$, where a third 'disinhibitory ring' appears. c) Illustration of the widening of the periphery due to a reduction in $1/1.2$ of the resolution from center to periphery.

and their extension to the continuum to Hermite Functions and Functionals) are extensively treated in [9]. They are a powerful tool in modular distributed computation but we shall focus here only on those properties relevant to retinal modelling.

Figure 2 shows the profile of several Newton Filters. First, there is a more than apparent similitude of these profiles with the spatial weighing functions reported for different retinal cells. However, much more important is that they are the result of some type of simple "dendritic like" microcomputation produced by the simplest types of excitation (addition) and inhibition (subtraction). Second, the shapes of the receptive fields are independent of where the excitation and inhibition rows are. Third, as results on generalized Newton Filters show, any

type of linear filter can be embodied by these structures if weighted addition and subtraction in each row is allowed (by weighted we mean, weights other than +1 or -1 in each node of the row, the same for each row).

It is apparent that one should accept that a neuron with a more complex dendritic arborization should present more complex behavior (or, simply, a more complex type of computation). This quasi-principle was the one which compelled Lettvin and co-authors to identify the Group two ganglion cells (their "bug detectors") as the "many level E shaped dendritic tree cell" in Ramon y Cajal's drawings. The microcomputational scheme provided by Newton's filters "dendrites" reiterates the fact, even though we are still considering only linear processes. This could also provide for a computational "explanation" of why that, and not something else, giving similar qualitative results (such as Bessel functions or Fourier peripherally attenuated transforms), may be present in the retina.

The set of all filters of a given length, L , is a complete set. There are L of said filters, which are obtained by giving to m, n in $N(A_m, D_n)$ the integer values $0, 1, \dots, L-1$, subjected to the restriction $m+n=L-1$. The weights of the filter form a vector, and the set of all L vectors is an independent set. Thus, they provide for a complete set of $\{C_n\}$ descriptors to characterize any input.

When working with large receptive fields, the set $\{C_n\}$ may be truncated for practical purposes. Notice that a single computing cell may compute more than one C_n if there exists a mechanism, which is not difficult to imagine, which will change the nature of excitatory to inhibitory in some "dendritic rows". Also, via this mechanism, due to the overlapping of receptive fields, a smaller set of computing cells may convey almost the totality of the information arriving at this theoretical retina. This may be an acceptable computational explanation of the "multiple meaning" which Lettvin found in the frog's retinal ganglia some 30 years ago. It is clear that the output of a cell which computes several descriptors is fairly complicated to decipher which usually is the case for retinæ where there is a drastic reduction in the number of output fibers and yet a reasonable amount of local information needs to be preserved.

The extension of Newton's filters to the continuum allows for a better visualization of the "shapes" of receptive field weights. Normalized Newton's filter $N(A_m, D_0)/S$ tends to a Gaussian as $m \rightarrow \infty$ (S is the central value of the filter). Filters $N(A_m, D_1), N(A_m, D_2)$, etc. tend to the first, second, etc, derivatives of a Gaussian. In general,

$$N(A_m, D_n) \rightarrow (-1)^n H_n(x) e^{-x^2} = \phi(x)$$

where $H_n(x)$ are the Hermite Polynomials ($H_0 = 1, H_1 = 2x, H_2 = 4x^2 - 2$, etc.). $-N(A_m, D_2)$ (2) and its continuum representation, show the typical center-surround spatial organization. It is classic that neurophysiologists tend to approximate this type of receptive fields to a difference of two Gaussians, though $-\phi_2$ has a sounder basis. David Marr proposed, that a good filter for

detecting zero crossings in edge detection for low level computational vision, is precisely $-\phi_2$, though he did not identify it as one of the Hermite functions.

The main objection to Marr's proposal of the second derivative of a Gaussian as a spatial model for linear ganglia, comes from two facts: first the surrounding Off or inhibitory ring is much wider in real cells than it is predicted by said filter. Second, the filter is a high pass which eliminate all DC spatial components, which is not the case for cuasi-linear ganglia. These two objections are not so for Newton Filters, since they provide for local lateral computational structures that might be variable and flexible, instead of the unchangeable Hermite Functions. Thus, the first effect is explained because the resolution in the computing lateral interaction net is decreasing from center to periphery (which is reasonable), and the second because the weights are not exactly +1 or -1 (which is also reasonable). This is illustrated by the tool of the next section.

3.2 A Computational Tool for Lateral Interaction

We are developing a computer tool to provide both for the analysis and the synthesis of receptive fields in one dimension, where, given the microstructure of a receptive field, the weight pattern is found; and viceversa, find the microstructure 'dendritic' connections that produce a given receptive field weight profile. This tool has been demonstrated for a number of cases and is being extended to include time and non-linear functions to simulate a more general function response of the type of $F(I_{ijk})$, working on a spatio-temporal input space.

The main screen of this tool is shown in figure 3. It's a standard application for Microsoft Windows created with Delphi. The first parameter we can control is the number of inputs in one dimension, that is to say, the number of stimuli. If we change this parameter and press the button below, the grid just to the right of the edit window of the number of inputs, changes in size, allowing for the introduction of a greater, or lower, number of local weights.

Underneath the button that controls the number of inputs, appears two edit windows that allow for copying to the values of local weights from a layer to some others, facilitating the edition of the local weights. In the first edit window, we must introduce the layer to copy from, and the second window is for copying to the layer that we introduce. The first column in the grid shows the layer and the second one shows the local weights.

The local weights of each layer i can be written in the form a_i, b_i or in the fastest form $1, e_i$, where $e_i = \frac{b_i}{a_i}$. The grid where the user can introduce the local weights, can be completely erased with the button just below the grid. The grid can also be edited, that is to say, once a pair of values $1, e_i$ is introduced in a particular cell, can be changed later, clicking on it, and modifying its value.

The button labelled 'Newton' fires the computation of the global weights. The output appears in a grid just to the left of that button.

Once the global weights are calculated, the screen shows two graphics: one with the distribution of the global weights (in red), with a horizontal linear scale (separation of the stimuli) and the other one (in blue), with a nonlinear scale.

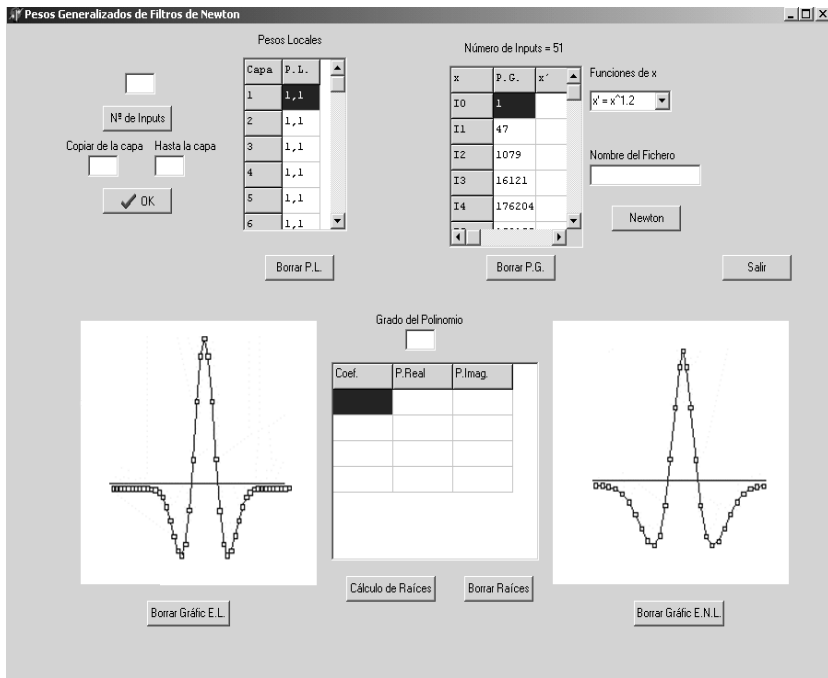


Fig. 3. View of the screen of the computer tool.

The horizontal non linear scale is controlled in the upper right part of the main screen in a 'combobox' with a set of predefined non linear functions.

This non linear scale accounts for the lowering of resolution from center to periphery, cited above.

The inverse problem, that is, given a weight profile find the local weights, is also solved by the tool. This implies finding the roots of a polynomial of degree n . In this case, the Jenkins-Traub algorithm [12] is used.

References

1. Troy, J.B., Shou, T.: The Receptive Fields of Cat Retinal Ganglion Cells in Physiological and Pathological States: Where We Are After Half a Century of Research. *Progress in Retinal and Eye Research* **21** (2002) 263–302
2. Enroth-Cugell, Ch., Lennie, P.: The Control of Retinal Ganglion Cells Discharge by Receptive Field Surround. *J. Physiol. (London)*. **247** (1975) 551–578
3. Winters, R.W., Hamasaki, D.I.: Temporal Characteristics of Peripheral Inhibition of Sustained and Transient Ganglion Cells in Cat's Retina. *Vision Research*, **16** 1976 37–45
4. Pearlman, A.L., Hughes, C.P., Functional Role of Efferents to the Avian Retina. *Comp. Neurol.* **166** (1976) 111–112

5. Bäckström, A.C., Reuter, T.: Receptive Field Organization of Ganglion Cells in the Frog. *J. Neurophysiol.* **246** (1975) 79–107
6. Mira, J., Moreno-Díaz, R.: Un Marco Teórico para Interpretar la Función Neuronal a Altos Niveles. In *Biocibernética: Implicaciones en Biología, Medicina y Tecnología*. Moreno-Díaz, R. and Mira, J. (Eds.), (1984) 149–171. Siglo XXI, Madrid, Spain.
7. Tessier-Lavigne, M.: Visual Processing in the Retina. In: *Principles of Neural Science*, 4th edition. Kandel, E.R., Schwartz, J.H., Jessell, T.M. (Eds.). McGraw-Hill, New York, (2000) 507–522
8. Marr, D.: *Vision*, WH Freeman, New York (1980)
9. Moreno-Díaz, R. (jr.): *Computación Modular Distribuida: Relaciones Estructura Función en Retinas*, PhD Thesis, Universidad de Las Palmas de Gran Canaria. Spain. (1993)
10. Hammond, P.: Contrasts in Spatial Organization of Receptive Fields at Geniculate and Retinal Levels: Centre, Surround and Outer Surround. *J. Physiol* **228** 115–137
11. Li, C.-Y., Zhou, Y.-X., Pei, X., Qiu, F.-T., Tang, C.-Q., Xu, X.-Z.: Extensive Disinhibitory Region Beyond the Classical Receptive Field of Cat Retinal Ganglion Cells. *Vision Res.* **32** 219–228
12. Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T.: *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed. Cambridge, England. Cambridge University Press, p.369 (1992)

A Novel Gauss-Markov Random Field Approach for Regularization of Diffusion Tensor Maps

Marcos Martín-Fernández¹, Raul San Josá-Estépar^{1,2}, Carl-Fredrik Westin²,
and Carlos Alberola-López^{1*}

¹ E.T.S. Ingenieros de Telecomunicación,
Universidad de Valladolid,
Cra. Cementerio s/n, 47011 Valladolid, SPAIN
{marcma, caralb}@tel.uva.es
rjosest@pi.tel.uva.es
² Surgical Planning Laboratory,
Brighman and Women's Hospital and
Harvard Medical School,
75 Francis St., Boston, MA 02115, USA
{rjosest,westin}@bwh.harvard.edu

Abstract. In this paper we propose a novel Gaussian MRF approach for regularization of tensor fields for fiber tract enhancement. The model follows the Bayesian paradigm: prior and transition. Both models are given by Gaussian distributions. The prior and the posterior distributions are Gauss-MRFs. The prior MRF promotes local spatial interactions. The posterior MRF promotes that local spatial interactions which are compatible with the observed data. All the parameters of the model are estimated directly from the data. The regularized solution is given by means of the Simulated Annealing algorithm. Two measures of regularization are proposed for quantifying the results. A complete volume DR-MRI data have been processed with the current approach. Some results are presented by using some visually meaningful tensor representations and quantitatively assessed by the proposed measures of regularization.

1 Introduction

Diffusion Tensor (DT) Magnetic Resonance Imaging (MRI) is a volumetric imaging modality in which the quantity assigned to each voxel of the volume scanned is not a scalar, but a tensor that describes local water diffusion. Tensors have direct geometric interpretations, and this serves as a basis to characterize local structure in different tissues. The procedure by which tensors are obtained can be consulted elsewhere [9]. The result of such a process is, ideally speaking, a 3×3 symmetric positive-semidefinite (PSD) matrix [4].

Tensors support information of the underlying anisotropy within the data. As a matter of fact, several measures of such anisotropy have been proposed out of tensors to make things easier to interpret; see, for instance, [1,9]. However,

* To whom correspondence should be addressed.

these measures rely of the ideal behavior of the tensors, which may be in some cases far from reality due to the presence of noise in the imaging process itself. As was pointed out in [7], periodic beats of the cerebro-spinal fluid and partial volume effects may add a non-negligible amount of noise to the data, and the result is that the hypothesis of PSD may not be valid. Authors are aware of this fact, so some regularization procedures have been proposed in the past [5,6,7,9].

In this paper we focus on regularization of DT maps using Markov Random Fields (MRFs); other regularization philosophies exist (see, for instance, [8] and [10] and references therein) although they will not be discussed in the paper. About MRFs we are aware of the existence of other Markovian approaches to this problem [6,7], in which the method presented is called by the authors the *Spaghetti model*. These papers propose an interesting optimization model for data regularization. However, some issues could be a matter of discussion. The authors build their MRF on the basis of a neighborhood system that may change through the optimization process, a fact that is not theoretically correct, though acceptable practical results may be obtained. In addition, the transition model used by the authors does not seem to have a clear probabilistic interpretation, but, in our opinion, only a functional interpretation.

The work we are about to present is an extended and detailed version of the short paper [3]. The model is built upon a Bayesian philosophy in which the two terms, namely, the prior and the likelihood function, have a clear physical meaning. Closed-form expressions have been obtained for the posterior, so the resulting model has a solid probabilistic foundation and, in our opinion, mathematical elegance. The maximum *a posteriori* (MAP) estimator will be found by means of the Simulated Annealing algorithm [2].

The paper is structured as follows. In section 2 we provide some background information on MRFs. This section provides most of the terminology that will be used in the paper. In section 3 we present the Bayesian model upon which the regularization process will be carried out. Then, section 4 proposes two quantitative measures of roughness that will be used to evaluate the regularization achieved by the proposed method. In section 5 several results are presented and, finally, in section 6 we present some conclusions and several issues that are still unaddressed.

2 Background on MRFs

A MRF models a multidimensional random variable with local (generally spatial) interactions. The MRF is defined on a finite grid or lattice $\mathbf{S}$ of sites (pixels or voxels). For a field to be a MRF, two conditions have to be satisfied:

- Positivity property: all the configurations are always possible, i.e., the probability density function must satisfy:

$$p(\mathbf{x}) > 0 \quad \forall \mathbf{x} \in \mathcal{R}^{|\mathbf{S}|}, \quad (1)$$

where $|\cdot|$ stands for the cardinality of its set argument.

- Markov property: the probability density function of one site conditioned to all the others sites must be equal to the probability density function of this site conditioned to its neighbors only, i.e.:

$$p(x_s/x_u, u \in \mathbf{S}, u \neq s) = p(x_s/x_u, u \in \delta(s)), \quad (2)$$

where $\delta(s)$ is a predefined fixed neighborhood for site s . Neighborhood systems must satisfy:

- $t \in \delta(s) \iff s \in \delta(t)$, and
- $s \notin \delta(s)$.

The conditional probability function $p(x_s/x_u, u \in \delta(s))$ is called the local characteristic of the field.

A multivariate Gaussian distribution defined over a lattice $\mathbf{S}$ with local spatial interactions is a MRF for the defined neighborhood system. This MRF so defined is called Gauss-MRF or autonormal distribution.

Typically the cardinality of $\mathbf{S}$ is too great so sampling the field directly is not feasible. The same problem occurs when the MAP estimation is sought; a *brute-force* search for the field mode is also unfeasible. Two algorithms, the convergence properties of which have been theoretically demonstrated, are used to solve each of the two problem, namely, the Gibbs Sampler algorithm, which provides field realizations, and the Simulated Annealing algorithm which provides field maximizers [2].

3 The Model

The model presented in this paper operates on each slice of the volume data; it will be applied independently to each component of the tensor field. We have only considered the 2D case, so the third row and the third column of each tensor have been discarded. However, as will be soon clear, the model here proposed carries over trivially to the 3D case.

A Bayesian framework consists of two ingredients: the prior and the transition (or the likelihood) models; these are connected in the posterior, the third probability model involved, by means of the Bayes theorem. The solution is some sort of parameter on the posterior (the mean, the median or the mode). In our case, we have resorted to a MAP solution, i.e., to finding the mode of the posterior.

Thus we have divided the exposition of the model in three subsections, as follows.

3.1 The Prior

Let $\mathbf{x} \in \mathbb{R}^{|\mathbf{S}|}$ denote a (non-observable) realization of the field consisting of one of the tensor components for every pixel; if the slice consists of M rows and N columns, vector $\mathbf{x}$ is a column vector with $M \times N$ components. The prior captures the knowledge about the field in the absence of any data. For smoothness to be

guaranteed, a reasonable assumption is to consider a Gauss-MRF or autonormal model:

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^{|\mathbf{S}|} \det(\boldsymbol{\chi})}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\kappa})^T \boldsymbol{\chi}^{-1} (\mathbf{x} - \boldsymbol{\kappa}) \right], \quad (3)$$

with $\boldsymbol{\kappa}$ and $\boldsymbol{\chi}$ the mean vector and covariance matrix respectively. The structure of latter must obey the MRF property. As previously stated, sampling this distribution directly is not a sensible choice. So, an alternative is to find the local characteristics which, in this case, turns out to be a one dimensional Gaussian distribution at site $s \in \mathbf{S}$ given by:

$$p(x_s/x_u, u \in \delta(s)) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp \left[-\frac{(x_s - \eta_s)^2}{2\sigma_s^2} \right], \quad (4)$$

for a given homogeneous neighborhood system $\delta(s)$ for each site $s \in \mathbf{S}$. The local characteristic parameters are the local mean η_s and the local variance σ_s^2 , which are functions of the neighboring sites $\delta(s)$ exclusively. This dependence will not be explicitly written to ease the notation. This prior parameters are estimated from those neighboring sites by means of the maximum likelihood (ML) method, which in the Gaussian case gives the common estimators:

$$\eta_s = \frac{1}{|\delta(s)|} \sum_{u \in \delta(s)} x_u \quad \text{and} \quad \sigma_s^2 = \frac{1}{|\delta(s)|} \sum_{u \in \delta(s)} (x_u - \eta_s)^2. \quad (5)$$

3.2 The Transition Model

In the transition model (or the noise model) the observed field $\mathbf{y} \in \mathbb{R}^{|\mathbf{S}|}$ is given by:

$$\mathbf{y} = \mathbf{x} + \mathbf{n}, \quad (6)$$

where $\mathbf{n} \in \mathbb{R}^{|\mathbf{S}|}$ is the noise field. The observed field is assumed to be a Gaussian field whose mean is equal to the non-observable field, thus, the probability density function of the observed field $\mathbf{y}$ conditioned to $\mathbf{x}$ is:

$$p(\mathbf{y}/\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^{|\mathbf{S}|} \det(\boldsymbol{\Upsilon})}} \exp \left[-\frac{1}{2} (\mathbf{y} - \mathbf{x})^T \boldsymbol{\Upsilon}^{-1} (\mathbf{y} - \mathbf{x}) \right], \quad (7)$$

with $\boldsymbol{\Upsilon}$ is the noise covariance matrix. This matrix is commonly accepted to be proportional to the identity matrix, so the noise field is conditionally white and stationary. Therefore, we can write for each pixel:

$$p(y_s/x_s, x_u, u \in \delta(s)) = p(y_s/x_s) = \frac{1}{\sqrt{2\pi}\sigma_n} \exp \left[-\frac{(y_s - x_s)^2}{2\sigma_n^2} \right]. \quad (8)$$

The noise variance has to be estimated from the observed noisy data $\mathbf{y}$. Two possible solutions can be considered:

- The minimum of the local ML estimated variances:

$$\sigma_1^2 = \min_{s \in \mathbf{S}} \left[\frac{1}{|\delta(s)|} \sum_{u \in \delta(s)} \left(y_u - \frac{1}{|\delta(s)|} \sum_{t \in \delta(s)} y_t \right)^2 \right]. \quad (9)$$

- The mean of the local ML estimated variances:

$$\sigma_2^2 = \frac{1}{|\mathbf{S}|} \sum_{s \in \mathbf{S}} \left[\frac{1}{|\delta(s)|} \sum_{u \in \delta(s)} \left(y_u - \frac{1}{|\delta(s)|} \sum_{t \in \delta(s)} y_t \right)^2 \right]. \quad (10)$$

We have observed that the minimum σ_1^2 tends to underestimate the value, while the mean σ_2^2 overestimates it; a value in between have been considered a good choice.

3.3 The Posterior Model

Bayes theorem allows us to write:

$$p(\mathbf{x}/\mathbf{y}) = \frac{p(\mathbf{y}/\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}, \quad (11)$$

where we know that:

$$p(\mathbf{y}) = \int_{\mathbb{R}^{|\mathbf{S}|}} p(\mathbf{y}/\mathbf{x})p(\mathbf{x})d\mathbf{x}. \quad (12)$$

The distribution $p(\mathbf{x}/\mathbf{y})$ is known to be a Gauss-MRF with joint density function:

$$p(\mathbf{x}/\mathbf{y}) = \frac{1}{\sqrt{(2\pi)^{|\mathbf{S}|}\det(\mathbf{A}_{\mathbf{y}})}} \exp \left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\xi}_{\mathbf{y}})^T \mathbf{A}_{\mathbf{y}}^{-1}(\mathbf{x} - \boldsymbol{\xi}_{\mathbf{y}}) \right], \quad (13)$$

with $\boldsymbol{\xi}_{\mathbf{y}}$ and $\mathbf{A}_{\mathbf{y}}$ the mean and the covariance matrix of the posterior, respectively. The dependence on the observed field is explicitly noted.

The MAP gives an estimation of the non-observable regularized field $\mathbf{x}$ given the noisy observation $\mathbf{y}$. As mentioned in the section 2, the Simulated Annealing algorithm [2] gives a practical procedure to find the MAP; to that end, the only piece of information that we need is the posterior local characteristic. This can be easily obtained by:

$$p(x_s/y_s, x_u, u \in \delta(s)) = \frac{p(y_s/x_s, x_u, u \in \delta(s))p(x_s/x_u, u \in \delta(s))}{p(y_s/x_u, u \in \delta(s))}, \quad (14)$$

where we know that:

$$p(y_s/x_u, u \in \delta(s)) = \int_{\mathbb{R}} p(y_s/x_s, x_u, u \in \delta(s))p(x_s/x_u, u \in \delta(s))dx_s. \quad (15)$$

Since both the prior and the likelihood are Gaussian, so is the posterior, thus the posterior local characteristic at site $s \in \mathbf{S}$ is:

$$p(x_s/y_s, x_u, u \in \delta(s)) = \frac{1}{\sqrt{2\pi}\rho_s} \exp \left[-\frac{(x_s - \mu_s)^2}{2\rho_s^2} \right], \quad (16)$$

where μ_s and ρ_s^2 are the posterior local mean and variance, which are given, after some algebra, by:

$$\mu_s = \frac{\sigma_s^2 y_s + \sigma_n^2 \eta_s}{\sigma_s^2 + \sigma_n^2} \quad \text{and} \quad \rho_s^2 = \frac{\sigma_s^2 \sigma_n^2}{\sigma_s^2 + \sigma_n^2}. \quad (17)$$

Finally, to implement the Simulated Annealing algorithm a partially parallel visiting scheme and a logarithmic cooling schedule for the temperature T are adopted, thus fixing $\rho_s^2(T) = T\rho_s^2$ (the implementation details of this algorithm can be consulted elsewhere [2]). The Simulated Annealing algorithm is applied in parallel to the three distinguishable components of the symmetric matrix tensor. After the algorithm visits each site simultaneously for the three components, the PSD condition is tested. If this test is passed the algorithm proceeds with the next site, otherwise, the sample is discarded and generated again. In practice, this event rarely occurs.

4 Regularization Measures

Two measures of regularization based on local roughness have been proposed to quantify the results achieved:

- The Frobenius norm of tensor differences, defined by:

$$R_f = \sum_{s \in \mathbf{S}} \sum_{u \in \delta'(s)} \|\mathbf{A}_s - \mathbf{A}_u\|, \quad (18)$$

where $\mathbf{A}_s$ is the tensor matrix for site s and for the neighborhood system $\delta'(s)$ which is different from the model neighborhood system $\delta(s)$ so that regularization results are measured with neighbors not involved in the regularization process.

- The entropy of discretized tensor linear component, defined by:

$$R_e = - \sum_{m=1}^M \sum_{n=1}^N p_{mn} \log_2 p_{mn}, \quad (19)$$

where the joint probabilities p_{mn} are to be estimated from data. They are defined by:

$$p_{mn} = p(\Gamma = \Gamma_m, \Theta = \Theta_n), \quad (20)$$

where Γ_m and Θ_n are the M and N discrete possible values for the random variables Γ and Θ . This variables are defined as:

$$\Gamma = \begin{cases} \frac{c_s}{c_u} & c_s \leq c_u \\ \frac{c_u}{c_s} & c_s > c_u \end{cases} \quad \text{and} \quad \Theta = \angle \mathbf{v}_s - \angle \mathbf{v}_u, \quad (21)$$

where the $u \in \delta'(s)$ are any pair of neighboring sites. $\mathbf{v}_s$ is the main eigenvector of the tensor matrix $\mathbf{A}_s$ for site s . c_s represents the linear component of the tensor defined as [9]:

$$c_s = \frac{\lambda_{\max_s}}{\lambda_{\max_s} - \lambda_{\min_s}}, \quad (22)$$

with $\lambda_{\max_s}$ and $\lambda_{\min_s}$ the maximum and minimum eigenvalues of the tensor matrix $\mathbf{A}_s$. The I random variable ranges in the interval $(0, 1)$ while the Θ random variable in the interval $(0, \pi)$.

5 Some Experimental Results

We have processed a DT-MRI volumetric data of size $185 \times 70 \times 20 \times 2 \times 2$, i.e., twenty sections of size 185×70 . The model neighborhood system $\delta(s)$ is fixed to the 12 closer sites surrounding s and the neighborhood system $\delta'(s)$ is set to the difference set between the 20 closer sites and the 12 closer sites surrounding s for the quantitative measures of roughness. For the noise variance we have chosen three values in the interval (σ_1^2, σ_2^2) by fixing:

$$\sigma_n^2 = k(\sigma_2^2 - \sigma_1^2) + \sigma_1^2, \quad (23)$$

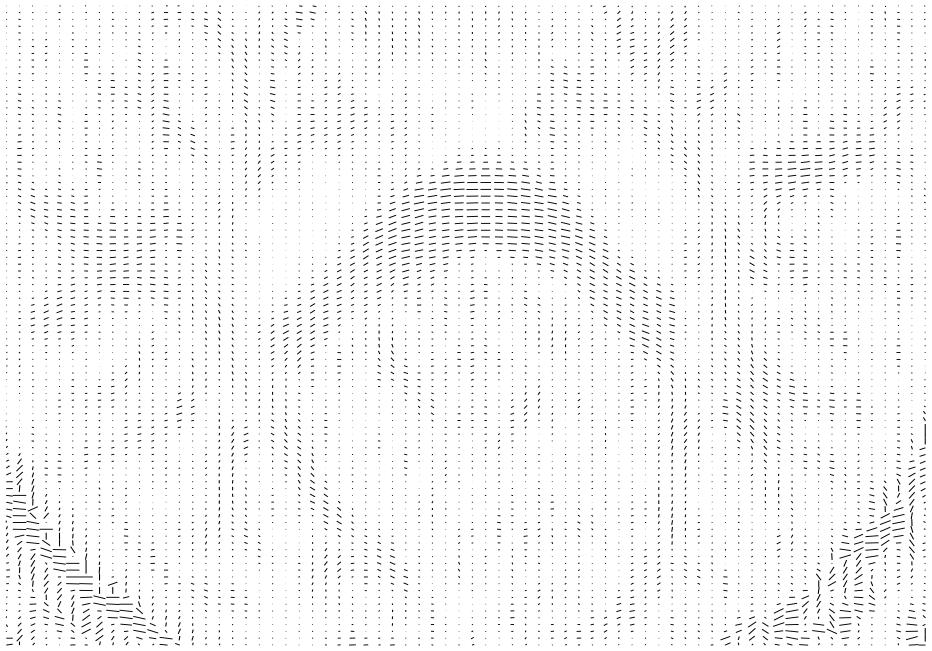
with $k \in (0, 1)$, yielding:

1. Low variance and thus low regularization, with $k = 0.25$.
2. Moderate variance and regularization, with $k = 0.5$.
3. High variance and regularization, with $k = 0.75$.

Figure 1(a) shows a detail of the original tensor map for slice number 20. Every vector is shown as a segment at site s , the angle and the length of which are given by the steering of the main eigenvector $\mathbf{v}_s$ and linear component c_s , respectively. In figure 1(b) a detail of the regularized tensor map is shown for the moderate regularization. It can be seen that the background noise is removed and the regions are more homogeneous without blurring the boundaries.

In figure 2(a) the Frobenius norm of the original tensor matrices for slice 20 is shown as an image in which the intensity of pixel s is given by $\|\mathbf{A}_s\|$. In figure 2(b) the same image representation is given for the moderate regularization. Comparing both images, the regularization achieved is perhaps now clearer than in figures 1(a) and 1(b). This leads us to propose this kind of image representation as a good procedure for visually assessing regularization results.

In figure 2(c) another representation is also proposed for the original tensor data for slice 20. In this case the image is a color coding of the image in figure 1(a) for which the RGB coordinated for site s are given by $R_s = \gamma c_s R(\angle \mathbf{v}_s)$, $G_s = \gamma c_s G(\angle \mathbf{v}_s)$ and $B_s = \gamma c_s B(\angle \mathbf{v}_s)$, with γ a gain factor (currently set to 2) and periodic functions $R(\cdot)$, $G(\cdot)$ and $B(\cdot)$ with period π so that the mapping between colors and angles is one-to-one. With this coding, the darker the pixel, the lower the linear component. In figure 2(d) the same color-coded



(a)



(b)

Fig. 1. (a) Original tensor map for slice number 20 (detail). (b) Regularized tensor map for slice number 20 (detail).

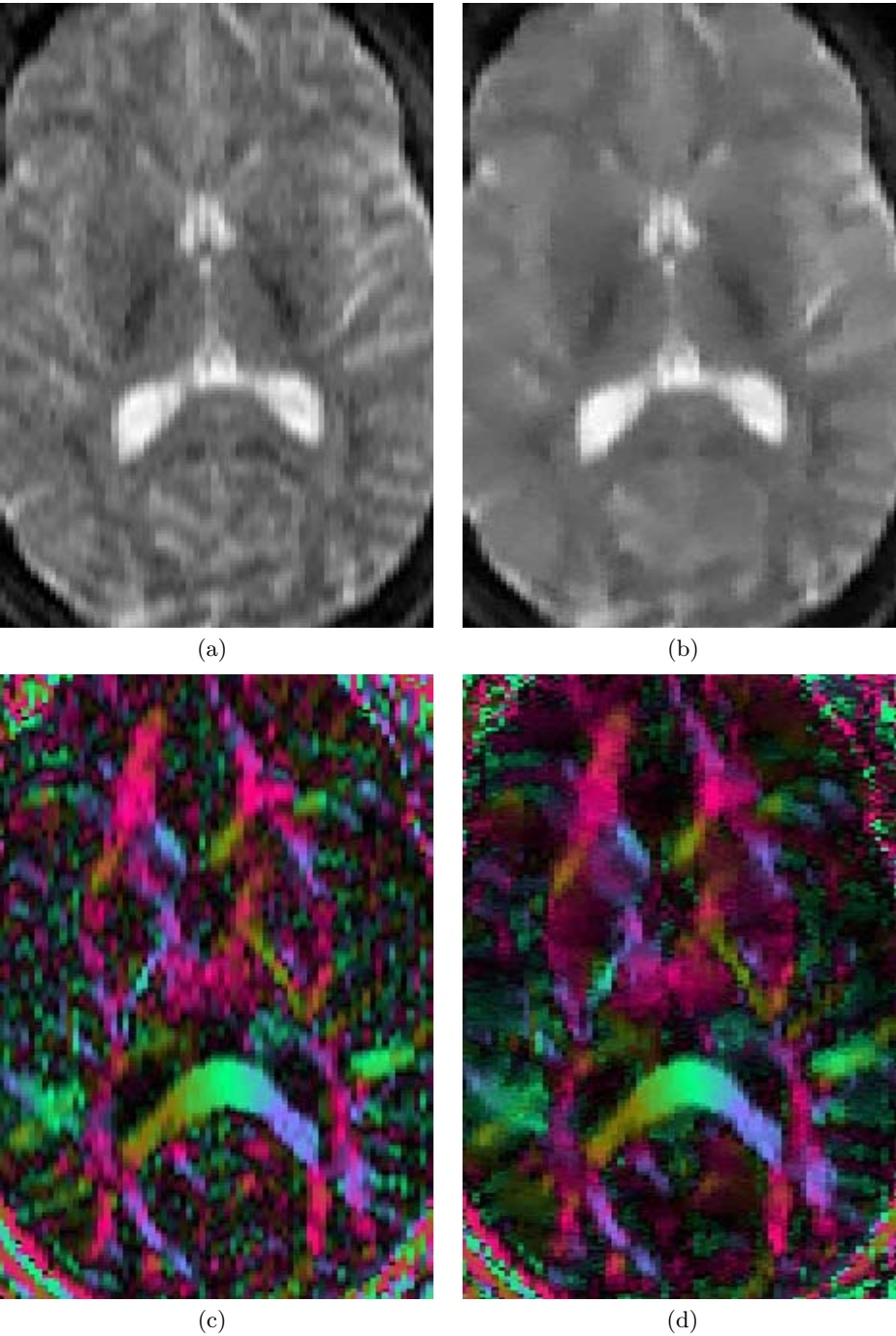
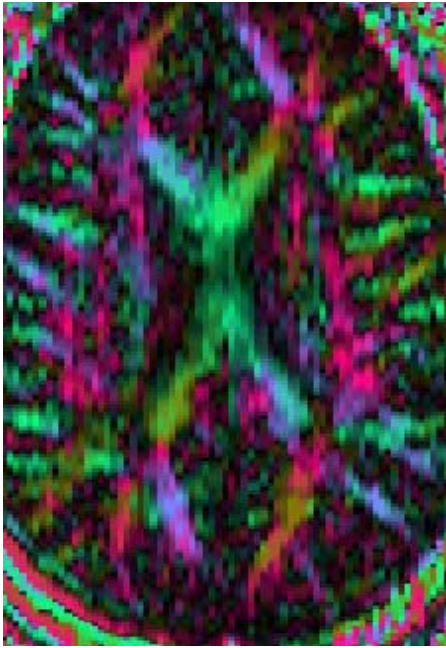
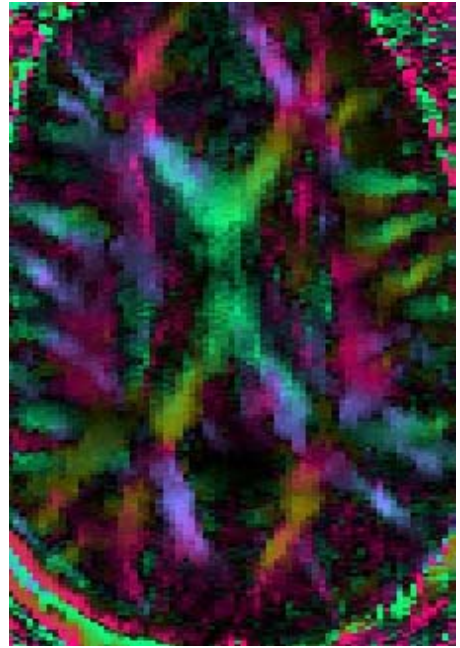


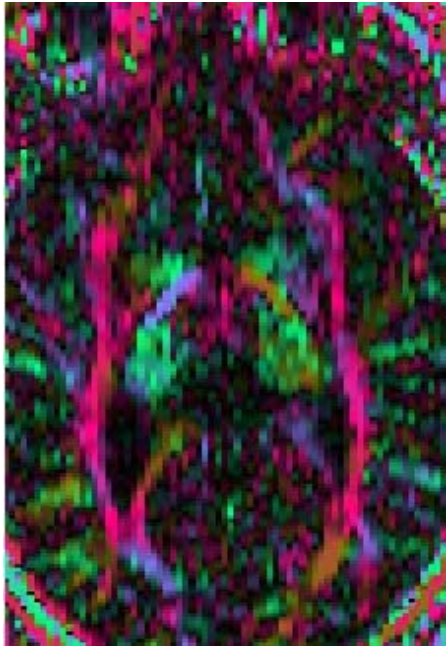
Fig. 2. For slice number 20: (a) original Frobenius norm image, (b) regularized Frobenius norm image, (c) original color-coded image, and (d) regularized color-coded image.



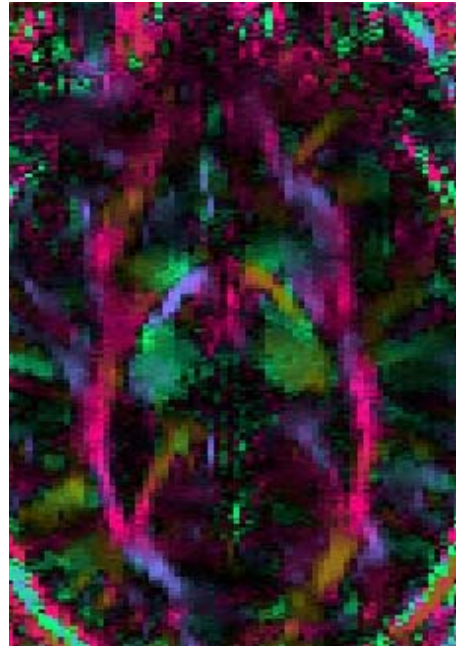
(a)



(b)



(c)



(d)

Fig. 3. For slice number 7: (a) original color-coded image, and (d) regularized color-coded image. For slice number 12: (c) original color-coded image, and (d) regularized color-coded image.

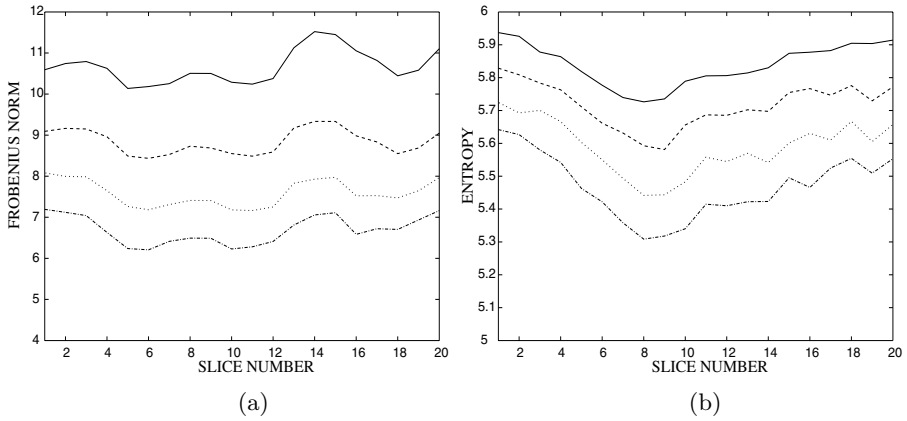


Fig. 4. Quantitative measures of regularization for the twenty slices with several noise variances: (a) Frobenius norm of tensor differences and (b) entropy.

image representation is given for the moderate regularization applied on slice 20. In this case the same comments as before apply.

The images shown in figures 3(a)-(b) and in figures 3(c)-(d) are analogous to the ones shown in figures 2(c)-(d) but in this case they represent the original and the moderate regularized color-coded images for slices 7 and 12, respectively.

We have calculated the two quantitative regularization measures proposed in section 4 for the 20 slices and for the three degrees of regularization. The results achieved are shown in figures 4(a)-(b). Figure 4(a) plots the Frobenius norm of tensor differences, the R_f value, as a function of the number of slice and parameterized by the degree of regularization. The lower curve is for high, the second for moderate and the third for low regularization respectively. The upper curve is for the original data. In figure 4(b) the entropy, the R_e value, is also plotted versus the slice number and parameterized by the degree of regularization with the same sorting scheme. The results obtained show clearly in a quantitative manner the degree of regularization which can be achieved with the presented approach.

6 Conclusions

In this paper we have described a novel probabilistic Bayesian model for the regularization of DT maps. The proposed model uses Gauss-MRFs for modeling the spatial relation of each component of the tensors along the field. All the model parameters are estimated directly from the data. A volumetric data set of twenty slices has been regularized by using the proposed method giving good results, the quality of which have been both visually assessed and quantified by the proposed measures of roughness. Also, color-coded images have been introduced to visualize the linear component of the tensors. Finally, the Frobenius norm

images are an alternative procedure to visually check the level of regularization that is achieved.

Some issues are still unaddressed:

- Exploiting the existing correlation between tensor elements.
- Studying the sensitivity with respect to the size of the neighborhood, both for the model and for the measures of regularization.
- Developing a procedure to guarantee the PSD condition without discarding samples in the optimization process.

Acknowledgments. The authors acknowledge the Comisión Interministerial de Ciencia y Tecnología for research grant TIC2001-3808-C02, the Junta de Castilla y León for research grant VA91/01, NIH grant P41-RR13218 and CIMIT.

References

1. P. Basser, C. Pierpaoli, Microstructural and Physiological Features of Tissues Elucidated by Quantitative-Diffusion-Tensor MRI, *Journal of Magnetic Resonance*, Ser. B, Vol. 111, No. 3, June 1996, pp. 209–219.
2. S. Geman, D. Geman, Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 6, No. 6, Nov. 1984, pp. 721–741.
3. M. Martín-Fernández, R. San José Estépar, C.F. Westin, C. Alberola-López, A Novel Gauss-Markov Random Field Approach for Regularization of Diffusion Tensor Maps, *Proc. of the NeuroImaging Workshop, Eurocast 2003*, Las Palmas de Gran Canaria, Feb. 2003, pp. 29–32.
4. G.H. Golub, C.F. Van Loan, Matrix Computations, *John Hopkins University Press*, Baltimore, Maryland, USA, 1996.
5. G.J.M. Parker, J.A. Schnabel, M.R. Symms, D.J. Werring, G.J. Barker, Nonlinear Smoothing for Reduction of Systematic and Random Errors in Diffusion Tensor Imaging, *Journal of Magnetic Resonance Imaging*, Vol. 11, No. 6, 2000, pp. 702–710.
6. C. Poupon, J.F. Mangin, V. Frouin, J. Regis, F. Poupon, M. Pachot-Clouard, D. Le Bihan, I. Bloch, Regularization of MR Diffusion Tensor Maps for Tracking Brain White Matter Bundles, in *Lecture Notes in Computer Science*, W. M. Wells, A. Colchester, S. Delp, Eds., Vol. 1446, Oct. 1998, pp. 489–498.
7. C. Poupon, C.A. Clark, V. Frouin, J. Regis, D. Le Bihan, I. Bloch, J.F. Mangin, Regularization Diffusion-Based Direction Maps for the Tracking of Brain White Matter Fascicles, *NeuroImage*, Vol. 12, 2000, pp. 184–195.
8. D. Tschumperlé, R. Deriche, DT-MRI Images: Estimation, Regularization and Application, *Proc. of the NeuroImaging Workshop, Eurocast 2003*, Las Palmas de Gran Canaria, Feb. 2003, pp. 46–47.
9. C.F. Westin, S.E. Maier, H. Mamata, A. Nabavi, F.A. Jolesz, R. Kikinis, Processing and Visualization for Diffusion Tensor MRI, *Medical Image Analysis*, Vol. 6, No. 2, June 2002, pp. 93–108.
10. C.F. Westin, H. Knuttsen, Tensor Field Regularization using Normalized Convolution, *Proc. of the NeuroImaging Workshop, Eurocast 2003*, Las Palmas de Gran Canaria, Feb. 2003, pp. 67–70.

Coloring of DT-MRI Fiber Traces Using Laplacian Eigenmaps

Anders Brun¹, Hae-Jeong Park², Hans Knutsson³, and Carl-Fredrik Westin¹

¹ Laboratory of Mathematics in Imaging, Brigham and Women's Hospital,
Harvard Medical School, Boston, USA,
`{anders,westin}@bwh.harvard.edu`

² Clinical Neuroscience Div., Lab. of Neuroscience,
Boston VA Health Care System-Brockton Division,
Dep. of Psychiatry, Harvard Medical School
and Surgical Planning Laboratory,
Brigham and Women's Hospital, Harvard Medical School,
`hjpark@bwh.harvard.edu`

³ Medical Informatics, Linköping University,
Inst. för medicinsk teknik, Universitetssjukhuset,
Linköping, Sweden
`knutte@imt.liu.se`

Abstract. We propose a novel post processing method for visualization of fiber traces from DT-MRI data. Using a recently proposed non-linear dimensionality reduction technique, Laplacian eigenmaps [3], we create a mapping from a set of fiber traces to a low dimensional Euclidean space. Laplacian eigenmaps constructs this mapping so that similar traces are mapped to similar points, given a custom made pairwise similarity measure for fiber traces. We demonstrate that when the low-dimensional space is the RGB color space, this can be used to visualize fiber traces in a way which enhances the perception of fiber bundles and connectivity in the human brain.

1 Introduction

Diffusion Tensor MRI (DT-MRI) makes it possible to non-invasively measure water diffusion, in any direction, deep inside tissue. In fibrous tissue such as muscles and human brain white matter, water tend to diffuse less in the directions perpendicular to the fiber structure. This means that despite the fact that spatial resolution in MRI is too low to identify individual muscle fibers or axons, a macroscopic measure of diffusion in a voxel may still reveal information about the fiber structure in it. Using DT-MRI it is therefore possible to infer the direction of the fiber in for instance white matter in the human brain. In particular, it is possible to estimate the direction of the fibers when the fiber organization is coherent within the voxel.

When a whole volume of data is acquired using DT-MRI, each voxel contains information about the local characteristics of diffusion inside that particular voxel. The diffusion is described by a tensor D , a symmetric positive definite 3×3 matrix, which

through the Stejskal-Tanner equation (1) explains the measurements obtained from the MR scanner

$$S_k = S_0 e^{-b \hat{g}_k^T D \hat{g}_k}. \quad (1)$$

Here $\hat{g}_k$ is a normalized vector describing the direction of the diffusion-sensitizing pulse, b is the diffusion weighting factor [11] and S_0 is a non-diffusion weighted measure. In order to estimate a tensor D inside each voxel, at least one non-diffusion weighted image S_0 and six diffusion weighted images with different directions are needed [18]. The product $\hat{g}_k^T D \hat{g}_k$ is often referred to as the Apparent Diffusion Coefficient, ADC, and describes the amount of diffusion in the gradient direction.

The tensor can be visualized as an ellipsoid, described by the eigenvectors of the diffusion tensor D , scaled with the square root of their respective eigenvalue. This ellipsoid will represent an isosurface of the probability distribution which describes the position of a water molecule, due to diffusion, a short time after it has been placed in the center of the tensor. A spherical ellipsoid therefore corresponds to an isotropic tensor, which describes that water diffusion is equally probable in any direction. When the ellipsoid is more oblate or elongated, it means that water diffuses less or more in a particular direction, and the tensor is therefore referred to as anisotropic. The anisotropy is often characterized using some rotationally invariant and normalized tensor shape measure, for instance the Fractional Anisotropy index [18]

$$FA = \frac{1}{\sqrt{2}} \frac{\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_1 - \lambda_3)^2}}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}. \quad (2)$$

One of the most intriguing uses of DT-MRI data is the possibility to follow and visualize fiber pathways in the brain. Traditionally this has been accomplished using fiber tracking algorithms, see for instance [1,2,18]. In these approaches a path originating from a seed point is calculated by iteratively moving a virtual particle in the direction in which diffusion is strongest, the principal diffusion direction (PDD). This direction corresponds to the major eigenvector of the diffusion tensor, which is the eigenvector corresponding to the largest eigenvalue. It is widely believed that for human brain white matter, in areas where the diffusion tensors are highly anisotropic, the PDD is highly correlated with the orientation of the underlying fiber structure.

One way to visualize the fiber organization of white matter is to place a virtual particle inside a voxel in white matter and iteratively move it according to a velocity field defined by the principal diffusion direction. This trace will be aligned with the underlying fiber structures and visualizing it will give the impression of looking at actual fiber pathways.

This paper will in the following sections introduce a novel post processing method for visualization of fiber traces from DT-MRI. We will focus on enhancing the perception of organization and connectivity in the data. The method will not specifically address the shortcomings of fiber tracking, but assume that a set of fiber traces has already been obtained. Instead the main contribution of this paper will be to show how a spectral non-linear dimensionality reduction technique, such as Laplacian eigenmaps, can be applied to the problem of organizing fiber trace data. The main application will be visualization of large collections of fiber traces.

2 Previous Work

Visualization of DT-MRI still poses a challenge for the medical imaging community, since the data is high dimensional and contains a lot of interesting anatomical structure. A simple but effective way to visualize tensor data is to map the tensors to scalars or colors and then visualize the data using any method for volume or image visualization. Commonly used scalar mappings include Fractional Anisotropy Index, trace and the norm of the tensor. Color mapping has also been used to encode orientation of the PDD. While these mappings are good in some applications, they are unintuitive or insufficient in others.

To cope with the high dimensionality of tensor data, special tensor glyphs have been designed, see for instance [18]. Commonly used glyphs are short line segments showing the orientation of the PDD and ellipsoids representing all six degrees of freedom of a tensor. Other interesting approaches to encode tensor shape and orientation are reaction diffusion patterns [10] and line integral convolution [12].

Fiber traces, as described in the introduction, have been successfully been used to reveal fiber pathways in the brain, see for instance [2]. Often the traces have been represented by streamtubes [19], sometimes in combination with coloring schemes and/or variation of the streamtube thickness according to some quality of the underlying tensor field.

In the area of post processing of fiber traces, prior to visualization, work on clustering of fiber traces have been reported recently. These approaches depend on a similarity measure between pairs of fiber traces, which is used in combination with a traditional clustering method (“fuzzy c-means clustering” [15] and “K nearest neighbors” [6]). Outside the medical field, model based curve clustering has been studied in [8]. The method presented in this article will share many similarities with automatic clustering methods. It will however give a continuous coloring of the fiber traces, as opposed to the discrete set of labels assigned during clustering. It could also be considered as a preprocessing step to clustering. Similar to the clustering methods, our approach is automatic and involves no user intervention except parameter selection. This is in sharp contrast from manual approaches to organize traces into bundles, such as the virtual dissection proposed in [5]. However, all the post processing methods for fiber traces share the same weakness: they rely on a good fiber tracking algorithm to perform well.

3 Embedding Fiber Traces – A Motivation

If fiber traces are initiated from seed points in the entire white matter, as in figure 1 left, a quick glance motivates the need for some kind of color mapping in order to enhance the perception of the fiber organization in the brain. We therefore propose a post processing step, prior to visualization, in which each fiber trace is assigned a color from a continuous RGB color space. The intuition is that similar traces should be assigned similar colors, while dissimilar traces are mapped to dissimilar colors. This will enhance the visualization of fiber bundles.

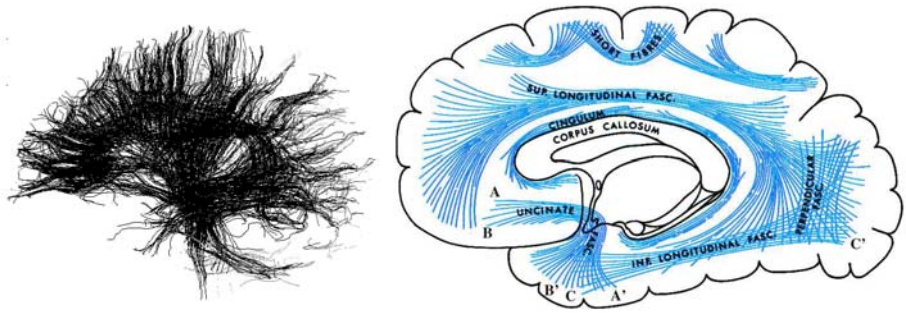


Fig. 1. Left: Fiber traces from a human brain. Simple PDD fiber tracking have been initiated from and constrained to voxels with high anisotropy index. A sagittal view. The head facing left. **Right:** A schematic view of major fiber bundles in the brain. Adapted from Gray's *Anatomy of the Human Body* as displayed at Bartleby.com.

4 Spectral Clustering and Embedding

In order to map the fiber traces we use a spectral embedding technique called Laplacian eigenmaps which was recently proposed by Belkin and Niyogi in [3]. The core of the algorithm is the use of a local similarity measure, which is used to construct a graph in which each node correspond to a data point and where the edges represent connections to neighboring data points. It is the structure of this graph which represents the manifold to be discovered, which is accomplished through the solution of an eigenvalue problem which maps each data point to a low-dimensional Euclidean space. This mapping locally preserves the graph structure. In short, points close in the graph are mapped to nearby points in the new Euclidean space.

In our application, the data points are fiber traces. The effect we would like to obtain is that traces within a fiber bundle are mapped to similar points in the low-dimensional space. The manifolds we hope to reveal would correspond to a parameterization of a specific fiber bundle. Not a parameterization along the fibers – all points of a fiber trace should project to the same point in the new low-dimensional space – but in the direction perpendicular to the fibers. In the case of a thin bundle such as the cingulate fasciculus we would expect a clustering effect to dominate, all traces within this thin bundle should project to more or less a single point in a low dimensional space. On the other hand, a large bundle structure such as the corpus callosum can be parameterized along the anterior-posterior axis and we would expect it to be represented as a one-dimensional manifold.

While fiber traces naturally reside in a low dimensional 3-D space, a trace itself must be considered as a high-dimensional object, or at least an object which we have difficulties in representing as a point in a low dimensional vector space. Constructing an explicit global similarity measure for fiber traces is also somewhat difficult – to what extent are two traces similar? How can we come up with a similarity measure which corresponds to a mapping of traces into a low-dimensional space? Luckily Laplacian eigenmaps and other spectral methods only needs a local similarity measure, a measure

which determine the similarity between a data point and its neighbors. This means that we only need to construct a similarity measure which is able to identify and measure similarity between two very similar traces. In the case of two very dissimilar traces, we may assume zero similarity.

Using this similarity measure, a graph is constructed in which nodes represent fiber traces and where edges connect neighboring traces.

5 Laplacian Eigenmaps

For an in depth explanation of Laplacian eigenmaps, as explained by Belkin and Niyogi, see [3]. In brief, the algorithm for Laplacian eigenmaps consists of three steps:

1. Construction of a graph where each node corresponds to a data point. Edges are created between nodes which are close to each other in the original space. A neighborhood of fixed size around each data point or the set of K nearest neighbors could for instance be used as criteria for creating the edges in the graph.
2. Weights are assigned to each edge in the graph. In general, larger weights are used for edges between points which are close to each other in the original space. In the simplest case, all weights are set to 1. A Gaussian kernel or similar could also be used.
3. Solution of the generalized eigenvalue problem:

$$D_{ij} = \begin{cases} \sum_{k=1}^N W_{ik} & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \quad (3)$$

$$L = D - W \quad (4)$$

$$Ly = \lambda Dy \quad (5)$$

where N is the number of nodes and L is called the Laplacian matrix of the graph. The eigenvectors derived from equation 3 are ordered according to their eigenvalues. Due to the structure of the L , the smallest eigenvalue will correspond to a constant eigenvector and is discarded, but the n eigenvectors corresponding to the next smallest eigenvalues are used as embedding coordinates for the data points in the new space.

We never performed the formation of the graph in step one explicitly, but performed a thresholding of the weights so that very small weights were set to zero, which corresponds to absence of an edge in the graph.

Laplacian eigenmaps share many similarities with other recent spectral algorithms for clustering and embedding of data, for instance Kernel PCA [14] and spectral methods for image segmentation [13], and we expect a qualitatively similar behavior from all of them even if the interpretation of the results is somewhat different in the various methods. For a unifying view of the behavior of spectral embeddings and clustering algorithms, see [4]. One of the most important aspects of spectral methods for clustering and embedding, including Laplacian eigenmaps, is the fact that they are all posed as eigenvalue problems, for which efficient algorithms are widely available.

6 Similarity Through Connectivity

There is no similarity measure given for fiber traces per se and therefore many ways of choosing the edge weights exist. In this initial effort to cluster and embed traces for visualization purposes, we will only try a simple but yet effective similarity measure.

The measure is based on the idea that two traces with similar end points should be considered similar. That is, we only look at the endpoints for a pair of fiber traces, and discard all other information. In figure 1 (right) we would for instance want a trace with endpoints $\{A, A'\}$ to have high similarity with a trace with endpoints $\{B, B'\}$. However, trace $\{C, C'\}$ should be considered dissimilar from both $\{A, A'\}$ and $\{B, B'\}$, even though they all share a common origin. This could also be interpreted as a measure of connectivity.

Here $f_{i,1}$ and $f_{i,\text{end}}$ corresponds to the first and last coordinates of the i th fiber trace and W_{ij} is the weight between nodes / fiber traces i and j :

$$f_i = (f_{i,1}, f_{i,\text{end}}), \quad (6)$$

$$\tilde{f}_i = (f_{i,\text{end}}, f_{i,1}), \quad (7)$$

$$W_{ij} = \begin{cases} 0 & \text{if } i = j \\ \exp\left(-\frac{\|f_i - f_j\|^2}{2\sigma^2}\right) + \exp\left(-\frac{\|f_i - \tilde{f}_j\|^2}{2\sigma^2}\right) & \text{if } i \neq j \end{cases} \quad (8)$$

We note that W_{ij} is symmetric with respect to i and j . This measure is also invariant to re-parameterization of the fiber trace, for instance reverse numbering the fiber trace coordinates. It will also give traces which connects similar points in space a large weight while dissimilar connectivity will result in a weight close to zero given that σ is chosen carefully.

This similarity measure will work fine in most cases where the fiber traces are not damaged and really connect different parts of the brain in an anatomically correct way. Other similarity measures used in clustering methods have been based on correlation measures between fiber traces [15,6]. Those correlation measures could be used as well to build up the graph needed by a spectral embedding method such as Laplacian eigenmaps. For the purpose of demonstration and under the assumption that the fiber traces are ok, the above described similarity should work fine and it is also faster to compute than correlation measures.

7 In vivo DT-MRI Data

Real DT-MRI data from the brain of a healthy volunteer was obtained at the Brigham and Women's Hospital using LSDI technique on a GE Signa 1.5 Tesla Horizon Echospeed 5.6 system with standard 2.2 Gauss/cm field gradients. The time required for acquisition of the diffusion tensor data for one slice was 1 min; no averaging was performed. The voxel resolution was $0.85\text{mm} \times 0.85\text{mm} \times 5\text{mm}$.

A random sample of 4000 points inside white matter with diffusion tensors having high FA were selected as seed points for the fiber tracking. Traces were then created by tracking in both directions starting from these seed points, following the principal eigenvector of diffusion using a step length of 0.5mm and linear interpolation of the

tensors. The tracking was stopped when reaching a voxel with FA lower than certain threshold approximately corresponding to the boundary between white and gray matter. Fiber traces shorter than 10mm were removed. This resulted in a set of approximately 3000 fiber traces.

8 Experiments

The algorithm was implemented in Matlab. While the number of fiber traces were at most 5000 the PDD tracking method, calculation of the graph Laplacian and the solution of the generalized eigenvalue problem could be performed without optimizations. Matlab was used for visualization except in figure 5, where the in-house software 3-D Slicer [9] was used.

For the color mapping, the The second, third and fourth eigenvector were scaled to fit into the interval $[0, 1]$ and then used for the channels red, green and blue, to color the corresponding fiber traces.

The embedding of fiber traces into a RGB color space was tested first on synthetic data, then on real human brain DT-MRI data. The synthetic toy examples should be considered as illustrations of the method rather than near realistic or challenging experiments.

Figure 2 shows how the embedding into color space works for a set of fiber traces arranged as a Möbius strip. The traces on (left) are mapped into an RGB space, which determines the color of each trace. In the right plots, the image of this mapping in RG space (first two embedding coordinates) is shown. Each dot in the right plots correspond to a single trace in the left plots. The circular structure of the Möbius strip can thus be seen in the geometry of the left image, in the coloring of the left image and in the shape of the fiber bundle after embedding it into RG space to the right.

Figure 3 (left) shows how traces connecting opposite sides of a sphere are colored. Coloring according to the three first embedding coordinates. This set of traces has the topology of the “projective plane”, P_2 . Note that even though it is impossible to see the traces inside the sphere, we can deduce how traces connect by looking at the colors which match on opposite sides. However, the projective plane cannot be embedded in three dimensions without intersecting itself, which means that the color mapping of this set of traces is many-to-one in some sense.

Figure 3 (right) shows a synthetic example of four fiber bundles, two crossing each other and two having the same origin. Because of our similarity measure based on connectivity of the fiber trace endpoints, crossings and overlaps will not disturb the embedding. Laplacian eigenmaps will color each in a color close to its neighbors colors. In this case the clustering properties of Laplacian eigenmaps becomes obvious, which is welcomed as no obvious manifold structure exists in the data.

The experiments on real data in figures 4 and 5 show how the method works in practice. The value of the only parameter σ was chosen empirically. Starting with a large sigma is safe in general, while a too small sigma give unstable solutions of the eigenvalue problem. In figure 5 an example is shown where the fiber traces have been projected back to a T2 weighted coronal slice.

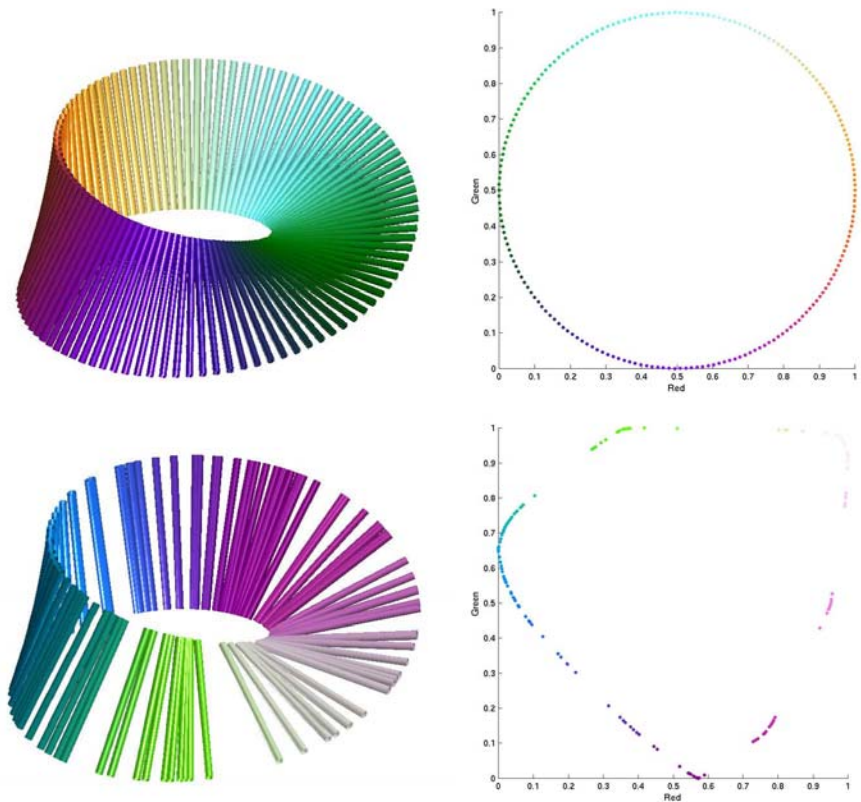


Fig. 2. Synthetic toy examples of coloring “fiber traces” shaped as a Möbius strip. **Top:** A very regular bundle (left) and its embedding (right). Note how the embedding finds a perfect circle. **Bottom:** A more random bundle (left) and its embedding using a little too small σ (right). Note how the embedding tends to enhance clusters in the data, but the topology is still somewhat a circle.

9 Discussion

All the figures show different aspects of the idea of using Laplacian eigenmaps, together with a custom made similarity measure, to enhance the visualization of fiber organization. Both the synthetic and real brain data show very promising results, and the colors reveal that the method has been able to organize and embed the fiber traces into a space where different anatomical structures have been mapped to different positions. In the real brain data, it can for instance be noted that traces on the left hemisphere in general have a different color from those on the right. Small structures such as the cingulum, going from posterior to anterior above the corpus callosum, are also more visible thanks to the coloring.

The experiments presented in this paper have been chosen with great care. Finding the correct σ has not always been easy and what is a good embedding of fiber traces in

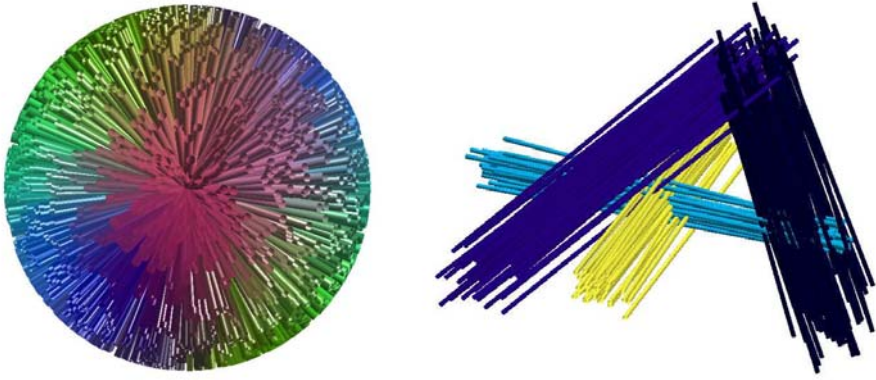


Fig. 3. Synthetic examples with “fiber traces” connecting opposite points on a sphere (left) and with four fiber bundles (right), two crossing each other and two having the same origin. Coloring according to the three first embedding coordinates.

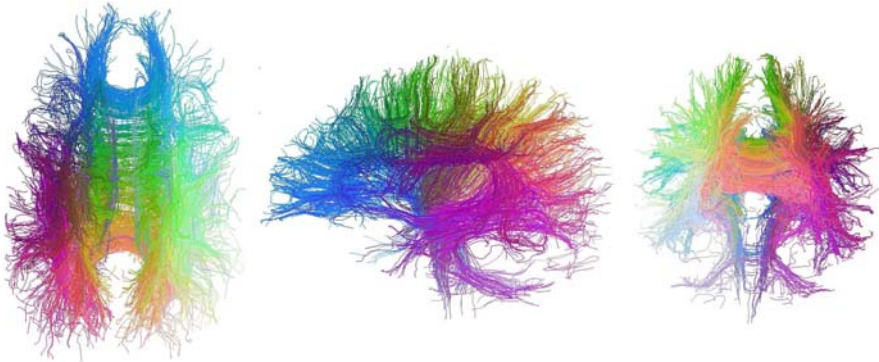


Fig. 4. Fiber traces from a human brain, colored such that traces with similar endpoints have been assigned similar colors. Simple PDD fiber tracking have been initiated from and constrained to voxels with high anisotropy index. **Left:** Axial view. The head facing up. **Middle:** Sagittal view. The head facing left. **Right:** Coronal view. The head facing inwards.

RGB-space for visualization is subjective. Optimal choice of σ as well as an analysis of the stability for the embedding is certainly interesting topics for future research.

We have so far not focused on optimizing the speed of this post processing method for fiber traces. After the coloring is only done once per dataset. However, for more than a maximum of 5000 fiber traces used in our experiments, we feel there is a need to take greater care in terms of memory management and speed. First of all the eigenvalue problem solved in Laplacian eigenmaps is sparse, given the right similarity measure.

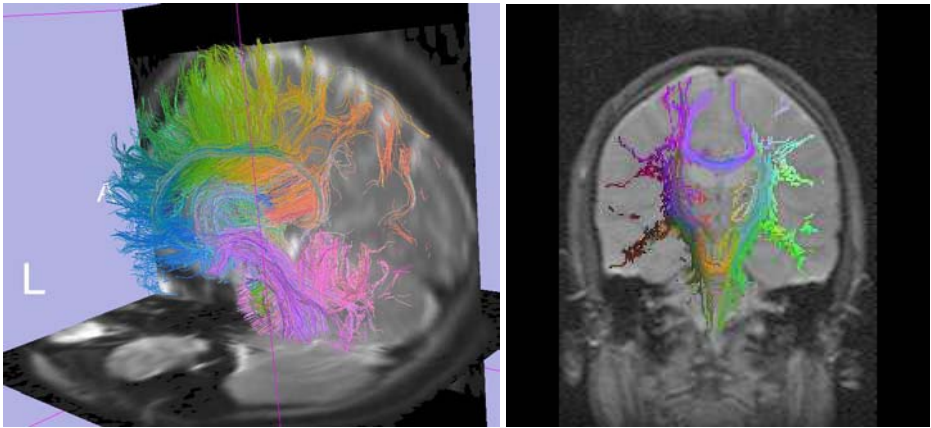


Fig. 5. Left: Fiber traces from a human brain, colored such that traces with similar endpoints are assigned similar colors. A cutting plane is used to give a cross section view of corpus callosum and slices from a T2 weighted volume add additional understanding of the anatomy. Visualization done using the 3-D Slicer [9]. **Right:** Fiber traces from a human brain, colored such that traces with similar endpoints are assigned similar colors. Only the intersection of the traces with a coronal T2 weighted slice is shown. This kind of voxel coloring could for instance assist when manually segmenting white matter in DT-MRI images.

Also there exists methods to reduce the size of the eigenvector calculation by using sampling methods such as in the Nyström method [7].

The similarity measure used so far is efficient, but simple. Correlation measures of fiber trace similarity have been used by other groups and this method for fiber trace visualization could definitely benefit from a better definition of local fiber trace similarity. Two issues raises. One is to define a better similarity measure which is able to “glue together” broken fiber traces, as fiber tracking is sensitive to noise. The other issue is speed, as the fiber trace similarity measure is evaluated for all pairs of traces. We have done experiments with highly efficient and more correlation-like similarity measures, but the results are still too preliminary to present here.

10 Conclusion

The goal of this project was to find a post processing method for DT-MRI fiber traces, to enhance the perception of fiber bundles and connectivity in the brain in general. We can conclude that despite the simplicity of the similarity function, this approach based on Laplacian eigenmaps has been able to generate anatomically interesting visualizations of the human brain white matter. Many interesting new topics arise in the light of this novel way of organizing DT-MRI data: clustering, segmentation and registration being prominent candidates for future research.

Acknowledgments. Thanks to Martha Shenton, Ron Kikinis, Stephan Maier, Steve Haker, Eric Pichon, Jonathan Sacks and Mats Björnemo for interesting discussions.

This work was funded in part by NIH grants P41-RR13218 and R01 MH 50747, and CIMIT. It was also supported in part by the Post-doctoral Fellowship Program of Korea Science & Engineering Foundation (KOSEF).

References

1. P. J. Basser Inferring microstructural features and the physiological state of tissues from diffusion-weighted images. *NMR in Biomedicine*. 8 (1995) 333–344.
2. P. J. Basser, S. Pajevic, C. Pierpaoli, J. Duda, and A. Aldroubi In vivo fiber tractography using DT-MRI data. *Magn. Reson. Med.*. 44 (2000) 625–632.
3. M. Belkin and P. Niyogi Laplacian eigenmaps and spectral techniques for embedding and clustering. in T. G. Diettrich, S. Becker, and Z. Ghahramani (eds:) *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA 2002.
4. M. Brand and K. Huang A unifying theorem for spectral embedding and clustering. 2003.
5. M. Catani, R. J. Howard, S. Pajevic, and D. K. Jones. Virtual in vivo interactive dissection of white matter fasciculi in the human brain. *Neuroimage* 17, pp 77–94, 2002.
6. Z. Ding, J. C. Gore, and A. W. Anderson. Classification and quantification of neuronal fiber pathways using diffusion tensor MRI. *Mag. Reson. Med.* 49:716–721, 2003.
7. C. Fowlkes, S. Belongie, and J. Malik. Efficient Spatiotemporal Grouping Using the Nyström Method *CVPR*, Hawaii, December 2001.
8. S. J. Gaffney and P. Smyth. Curve clustering with random effects regression mixtures. in C. M. Bishop and B. J. Frey (eds:) *Proc. Ninth Int. Workshop on AI and Stat.*, Florida, 2003.
9. D. Gering, A. Nabavi, R. Kikinis, W. Eric L. Grimson, N. Hata, P. Everett, F. Jolesz, and W. Wells III. An Integrated Visualization System for Surgical Planning and Guidance using Image Fusion and Interventional Imaging. *Proceedings of Second International Conference on Medical Image Computing and Computer-assisted Interventions, Lecture Notes in Computer Science*, pp. 809–819, 1999.
10. G. Kindlmann, D. Weinstein, and D. Hart Strategies for Direct Volume Rendering of Diffusion Tensor Fields *IEEE transactions on Visualization and Computer Graphics* Vol. 6, No. 2. 2000.
11. D. Lebihan, E. Breton, D. Lallemand, P. Grenier, E. Cabanis, and M. LavalJeantet MR imaging of intravoxel incoherent motions: application to diffusion and perfusion in neurologic disorders. *Radiology* 161, 401–407, 1986.
12. T. McGraw, B. C. Vemuri, Z. Wang, Y. Chen, and T. Mareci Line integral convolution for visualization of fiber tract maps from DTI *Proceedings of Fifth International Conference on Medical Image Computing and Computer-assisted Interventions*, Lecture Notes in Computer Science, pp. 615–622, 2003.
13. M. Meila and J. Shi A random walks view of spectral segmentation. In *AI and Statistics (AISTATS) 2001*, 2001.
14. B. Schölkopf, A. Smola, and K. Müller Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.
15. J. S. Shimony, A. Z. Snyder, N. Lori, and T. E. Conturo Automated fuzzy clustering of neuronal pathways in diffusion tensor tracking. *Proc. Intl. Soc. Mag. Reson. Med.* 10, Honolulu, Hawaii, May 2002.
16. E. Weisstein Eric Weisstein's world of mathematics. <http://mathworld.wolfram.com/>, May 15, 2003. .
17. C.-F. Westin, S. Peled, H. Gudbjartsson, R. Kikinis, and F. A. Jolesz Geometrical Diffusion Measures for MRI from Tensor Basis Analysis *Proc. of the International Society for Magnetic Resonance Medicine (ISMRM)*, Vancouver Canada, April 1997.

18. C.-F. Westin, S. E. Maier, B. Khidir, P. Everett, F. A. Jolesz, and R. Kikinis. Image Processing for Diffusion Tensor Magnetic Resonance Imaging *Proceedings of Second International Conference on Medical Image Computing and Computer-assisted Interventions, Lecture Notes in Computer Science*, pp. 641–652, 1999.
19. S. Zhang, T. Curry, D. S. Morris, and D. H. Laidlaw. Streamtubes and streamsurfaces for visualizing diffusion tensor MRI volume images. *Visualization '00 Work in Progress* October 2000.

DT-MRI Images : Estimation, Regularization, and Application

D. Tschumperlé and R. Deriche

Odyssée Lab, INRIA Sophia-Antipolis, France
{dtschump, der}@sophia.inria.fr
<http://www-sop.inria.fr/odyssee>

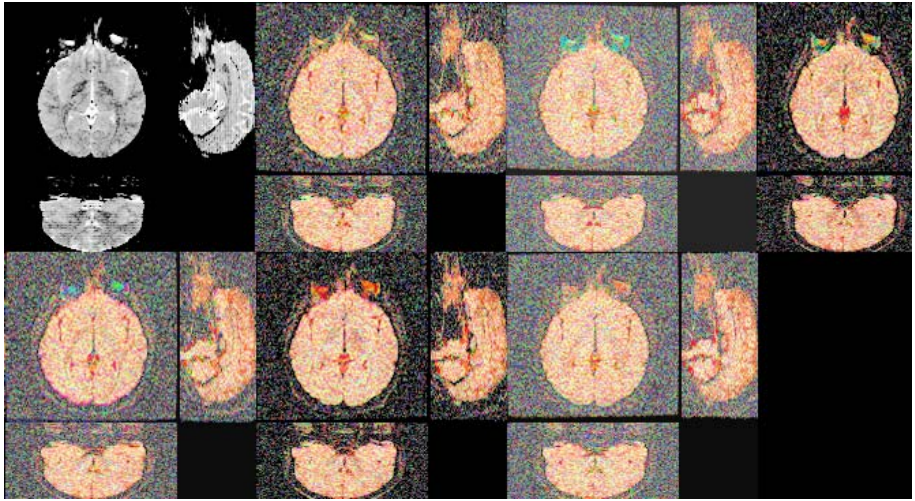
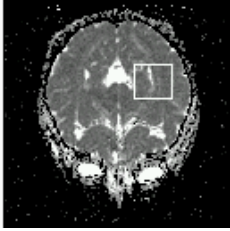
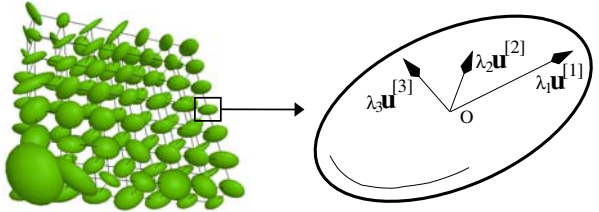
Abstract. Diffusion-Tensor MRI is a technique allowing the measurement of the *water molecule motion* in the tissues fibers, by the mean of rendering multiple MRI images under different oriented magnetic fields. This large set of raw data is then further estimated into a *volume of diffusion tensors* (i.e. 3×3 symmetric and positive-definite matrices) that describe through their spectral elements, the diffusivities and the main directions of the tissues fibers. We address two crucial issues encountered for this process : diffusion tensor *estimation* and *regularization*. After a review on existing algorithms, we propose alternative variational formalisms that lead to new and improved results, thanks to the introduction of important tensor constraint priors (positivity, symmetry) in the considered schemes. We finally illustrate how our set of techniques can be applied to enhance fiber tracking in the white matter of the brain.

1 Introduction

The recent introduction of DT-MRI (Diffusion Tensor Magnetic Resonance Imaging) has raised a strong interest in the medical imaging community [14]. This non-invasive 3D modality consists in measuring the water molecule motion within the tissues, using magnetic resonance techniques. It is based on the rendering of multiple raw MRI volumes $S_k : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{R}$ using pulse sequences with several gradient directions and magnitudes (at least 6 noncolinear directions are necessary). An additional image S_0 is also measured without preferred gradient direction (Fig.1a). These S_k may be quite noisy, due to the high speed needed for these multiple MRI acquisitions. This large set $\{S_k, k = 0 \dots n\}$ of raw volumes is then estimated into a corresponding volume $\mathbf{T} : \Omega \subset \mathbb{R}^3 \rightarrow \mathbf{P}(3)$ of Diffusion Tensors (i.e 3x3 symmetric and positive-definite matrices) that describe through their spectral elements, the main diffusivities $\lambda_1, \lambda_2, \lambda_3$ (with $\lambda_1 \geq \lambda_2 \geq \lambda_3$) and the corresponding principal orthogonal directions $\mathbf{u}^{[1]}, \mathbf{u}^{[2]}, \mathbf{u}^{[3]}$ of the water molecule diffusion in tissues such as bones, muscles and white matter of the brain (Fig.1b).

$$\forall x, y, z \in \Omega, \quad \mathbf{T}(x, y, z) = \lambda_1 \mathbf{u}^{[1]} \mathbf{u}^{[1]T} + \lambda_2 \mathbf{u}^{[2]} \mathbf{u}^{[2]T} + \lambda_3 \mathbf{u}^{[3]} \mathbf{u}^{[3]T}$$

Depending on the characteristics of the tissue, the diffusion (and then the estimated tensors) can be isotropic (for instance in the areas with fluids such

(a) Measured raw acquisitions S_k ($k = 0 \dots 6$)(b1) Mean diffusivity = $\frac{1}{3}(\lambda_1 + \lambda_2 + \lambda_3)$ (b2) Part of the estimated tensor field $\mathbf{T}$, represented with 3D ellipsoids.**Fig. 1.** Principle of DT-MRI Imaging

in the CSF filled ventricles) or anisotropic as in the white matter of the brain where the diffusion is mainly performed in the direction of the neuron fibers [15, 30]. DT-MRI is then particularly well adapted to study the brain connectivities, by tracking the fiber directions given pointwise by the principal eigenvector $\mathbf{u}^{[1]}(x, y, z)$ of the tensor $\mathbf{T}(x, y, z)$.

Actually, retrieving the fiber bundles from the raw images S_k involves two subjacent processes : First the *estimation part*, which estimates the diffusion tensors as gaussian models of the water diffusion, directly from the raw data S_k . Then, as the obtained tensor field $\mathbf{T}$ may be noisy, a specific *regularization process* can be necessary to improve the result.

Here we propose a survey of the related methods in the literature and introduce new variational frameworks that take important tensor structural constraints into account for these estimation and regularization steps. We highlight the different advantages of our formulations over the previous ones and we illustrate how our set of approaches can be used to obtain fiber tracking results from synthetic and real DT-MRI datasets of the brain.

2 Diffusion Tensor Estimation

2.1 Review of Existing Methods

The estimation process gathers the informations given by the multiple physical measures S_k of the diffusion MRI into a field of 3×3 symmetric matrices $\mathbf{T}$ which represent gaussian models of the water molecule diffusion. This link is given through the Stejskal-Tanner equation [21] :

$$\forall (x, y, z) \in \Omega, \quad S_{k(x,y,z)} = S_{0(x,y,z)} e^{-bg_k^T \mathbf{T}_{(x,y,z)} g_k} \quad (1)$$

where the b -factor is a constant, depending on the acquisition parameters and $g_k \in \mathbb{R}^3$ is a vector representing the pulse gradient magnitude ($\|g_k\|$) and direction ($g_k/\|g_k\|$) used for the acquisition of the image S_k . Classical methods for computing the tensor $\mathbf{T}$ from the images S_k have been already proposed in the literature :

- **Direct tensor estimation** : proposed by Westin-Maier [29], this method lies on the decomposition of $\mathbf{T}$ in an orthonormal tensor basis $\tilde{g}_k \tilde{g}_k^T$ (with $\tilde{g}_k = g_k/\|g_k\|$ and $k = 1..6$). The 6 coordinates of $\mathbf{T}$ in this basis (which are *dot products* in tensor space), naturally appear in eq.(1), and can then be retrieved :

$$\mathbf{T} = \sum_{k=1}^6 \langle \mathbf{T}, \tilde{g}_k \tilde{g}_k^T \rangle \tilde{g}_k \tilde{g}_k^T = \sum_{k=1}^6 \frac{1}{b\|g_k\|^2} \ln \left(\frac{S_0}{S_k} \right) \tilde{g}_k \tilde{g}_k^T \quad (2)$$

It particularly means that only 7 raw images $S_0, \dots, S_6$ are used to estimate the diffusion tensor field $\mathbf{T}$. As illustrated with the synthetic example in Fig.2, this low number of images may be not sufficient for a robust estimation of $\mathbf{T}$, particularly if the S_k are corrupted with a high variance noise.

- **Least square estimation**, is nowadays the most classical method used for diffusion tensor estimation since it can use all the available raw volumes S_k (see for instance [3,18]). The tensors $\mathbf{T}$ are estimated by minimizing the following least square criterion,

$$\min_{\mathbf{T} \in \mathcal{M}_3} \sum_{k=1}^n \left(\frac{1}{b} \ln \left(\frac{S_0}{S_k} \right) - g_k^T \mathbf{T} g_k \right)^2 \quad (3)$$

which leads to the resolution of an overconstrained system $\mathbf{A}\mathbf{x} = \mathbf{B}$ with a pseudo-inverse solution $\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B}$ (where $\mathbf{x}$ is a vector containing the six unknown coefficients $T_{xx}, T_{xy}, T_{xz}, T_{yy}, T_{yz}, T_{zz}$ of $\mathbf{T}$). The least square method generally gives better results for noisy datasets, since all the S_k (usually $n \gg 7$) are used in the estimation process.

Note that no one of both methods takes any prior positive-definite constraints of the tensors $\mathbf{T}$ into account. Nothing prevents the computation of negative

tensors (i.e with negative diffusivities). Practically, one has to check the tensor positivity after estimation, and reproject the negative tensors into the positive tensor space. This is generally done by forcing the negative eigenvalues of the tensors to zero : $\forall (x, y, z) \in \Omega, \tilde{\mathbf{T}} = \tilde{\lambda}_1 \mathbf{u}\mathbf{u}^T + \tilde{\lambda}_2 \mathbf{v}\mathbf{v}^T + \tilde{\lambda}_3 \mathbf{w}\mathbf{w}^T$, with $\tilde{\lambda}_i = \max(0, \lambda_i)$ (This projection minimizes the Mahalanobis distance between $\mathbf{T}$ and $\tilde{\mathbf{T}}$). Note also that both estimation methods are purely *pointwise* : no spatial interactions are considered during the estimation.

2.2 A Robust Variational Estimation

In order to avoid these important drawbacks, we propose a variational approach that estimates the tensor field $\mathbf{T}$ from the raw volumes S_k while introducing important priors on the *tensor positivity* and *regularity*. Our idea is based on the *positive-constrained minimization* of a least-square criterion, coupled with an anisotropic regularization term :

$$\min_{\mathbf{T} \in \mathbf{P}(3)} \int_{\Omega} \sum_{k=1}^n \left(\frac{1}{b} \ln \left(\frac{S_0}{S_k} \right) - g_k^T \mathbf{T} g_k \right)^2 + \alpha \phi(\|\nabla \mathbf{T}\|) d\Omega \quad (4)$$

where b is the constant factor depending on the acquisition parameters, g_k is the pulse gradient vector associated to the image S_k , $\alpha \in \mathbb{R}$ is a user-defined regularization weight and $\phi : \mathbb{R} \rightarrow \mathbb{R}$ is a regularizing ϕ -functional that measures the tensor field variations through the operator $\|\nabla \mathbf{T}\| = (\sum_{i,j} \|\nabla T_{i,j}\|^2)^{\frac{1}{2}}$. The minimization is then performed by a gradient descent (iterative method), *on the constrained space* $\mathbf{P}(3)$, representing the set of 3×3 symmetric and positive-definite matrices. Following our previous theoretical work on constrained matrix flows [9], the matrix-valued PDE minimizing (4) in $\mathbf{P}(3)$ with its natural metric is :

$$\begin{cases} \mathbf{T}_{(t=0)} = \mathbf{Id} \\ \frac{\partial \mathbf{T}}{\partial t} = -((\mathbf{G} + \mathbf{G}^T)\mathbf{T}^2 + \mathbf{T}^2(\mathbf{G} + \mathbf{G}^T)) \end{cases} \quad (5)$$

where $\mathbf{Id}$ is the 3×3 identity matrix and $\mathbf{G} = (G_{i,j})$ is the matrix defined as :

$$G_{i,j} = \sum_{k=1}^n \left(\frac{1}{b} \ln \left(\frac{S_0}{S_k} \right) - g_k^T \mathbf{T} g_k \right) (g_k g_k^T)_{i,j} - \frac{\alpha}{2} \operatorname{div} \left(\frac{\phi'(\|\nabla \mathbf{T}\|)}{\|\nabla \mathbf{T}\|} \nabla T_{i,j} \right)$$

Eq.(5) ensures the positive-definiteness of the tensors $\mathbf{T}$ for each iteration of the estimation process. Moreover, the regularization term α introduces some spatial regularity on the estimating tensor field, *while preserving important physiological discontinuities* thanks to the anisotropic behavior of the ϕ -function regularization formulation (as described in the broad literature on anisotropic smoothing with PDE's, see for instance [1,20,23,28] and references therein).

Moreover, a specific reprojection-free numerical scheme based on matrix exponentials can be used for this flow lying in $\mathbf{P}(3)$, as described in [9] :

$$\mathbf{T}_{(t+dt)} = \mathbf{A}^T \mathbf{T}_{(t)} \mathbf{A} \quad \text{with} \quad \mathbf{A} = \exp(-\mathbf{T}_{(t)}(\mathbf{G} + \mathbf{G}^T)dt)$$

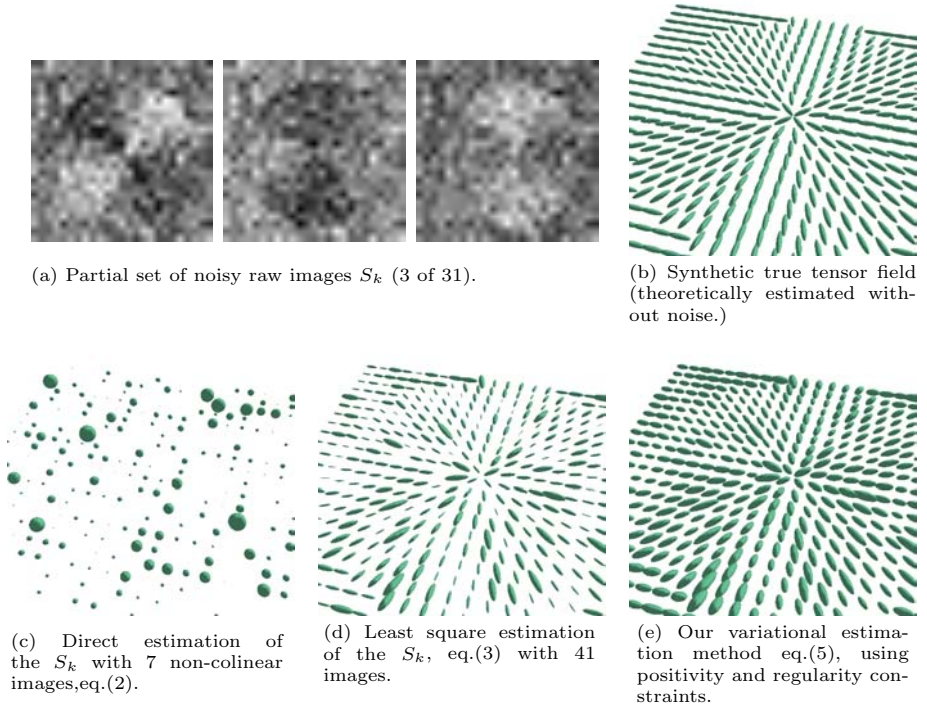


Fig. 2. DT-MRI Estimation : Comparison of our variational method for diffusion tensor estimation from noisy raw volumes S_k , with classical estimation techniques.

This scheme preserves numerically the positive-definiteness of the estimating tensors. The algorithm starts then at $t = 0$ with a field of *isotropic tensors* that are iteratively evolving in $P(3)$ until their shapes fit the measured data S_k with respect to the Stejskal-Tanner model eq.(1) and the positivity and regularity constraints. The respect of these natural diffusion tensor constraints has a large interest for DT-MRI, and leads to more accurate results than with classical methods. It is illustrated on Fig.2, with the estimation of a synthetic field from noisy images S_k .

In Fig.2d, notice the presence of false estimations, i.e negative estimated tensors that needed to be reprojected into the positive tensor space, and that appears very thin (at least one eigenvalue has been set to zero). These false estimations naturally disappear with our constrained method (Fig.2e). Raw data that tends to transform the positive tensors into negative ones are intrinsically ignored by the algorithm, thanks to the tensor positivity and regularity a-priori.

3 DT-MRI Regularization

During MRI image acquisition, the raw images may be corrupted by noise and specific regularization methods are needed to obtain more coherent diffusion

tensor maps. Recently, several methods have been proposed in the literature to deal with this important problem. These methods can be divided into two classes.

3.1 Non-spectral Regularization Methods

- **Smoothing the raw images S_k** : Vemuri-Chen-etal, proposed a scheme in [27] that regularizes directly the raw images S_k *before tensor estimation*, by using a PDE-based regularization scheme that takes the coupling between the S_k into account.

$$\forall k = 0 \dots n, \quad \frac{\partial S_k}{\partial t} = \operatorname{div} \left(\frac{g(\lambda_+, \lambda_-)}{\|\nabla S_k\|} \nabla S_k \right) - \mu(S_k - S_{k(t=0)})$$

The coupling here is done through the two eigenvalues $\lambda_{\pm}$ coming from a first estimation of the tensors $\mathbf{T}$, with a least square method. After regularization, the tensor field is re-estimated from the regularized version $\tilde{S}_k$, resulting in a smoother version of $\mathbf{T}$.

- **Direct matrix smoothing** : Another approach, proposed in [5,9] is to estimate the tensor field $\mathbf{T} : \Omega \rightarrow \mathcal{P}(n)$ from the S_k , then consider it as a multi-valued image with 6 components (i.e the number of different coefficients in a 3×3 matrix). This multivalued image is then processed with classic vector-valued diffusion PDE's (such as in [13,20,22,25,26]).

$$\frac{\partial T_{i,j}}{\partial t} = \operatorname{div} (\mathbf{D} \nabla T(i, j)) \quad (6)$$

where $\mathbf{D}$ is a 3×3 diffusion tensor that drives the regularization process. This tensor $\mathbf{D}$ generally depends on $\mathbf{T}$ and its spatial derivatives. Moreover, the method proposed in [9] ensures the tensor positivity constraint, in a theoretical way, as well as the respect of a natural metric in the positive tensor space.

- **Drawbacks** : By definition, non-spectral methods cannot have a direct control on the spectral elements of the tensors, which are however the relevant features that characterizes the biological tissues. During non-spectral regularization processes, tensor orientations and diffusivities are smoothed *at the same time*. Unfortunately, tensor diffusivities are regularizing faster than tensor orientation, resulting in an eigenvalue swelling effect for long time regularization (Fig.3). Then, a high risk of losing tensor orientation occurs : the tensors are quite fastly converging to *identity matrices*.

3.2 Spectral Regularization Methods

The idea behind spectral regularization methods of diffusion tensor fields lies in the separate (but eventually coupled) regularization of the tensor *diffusivities* λ_l

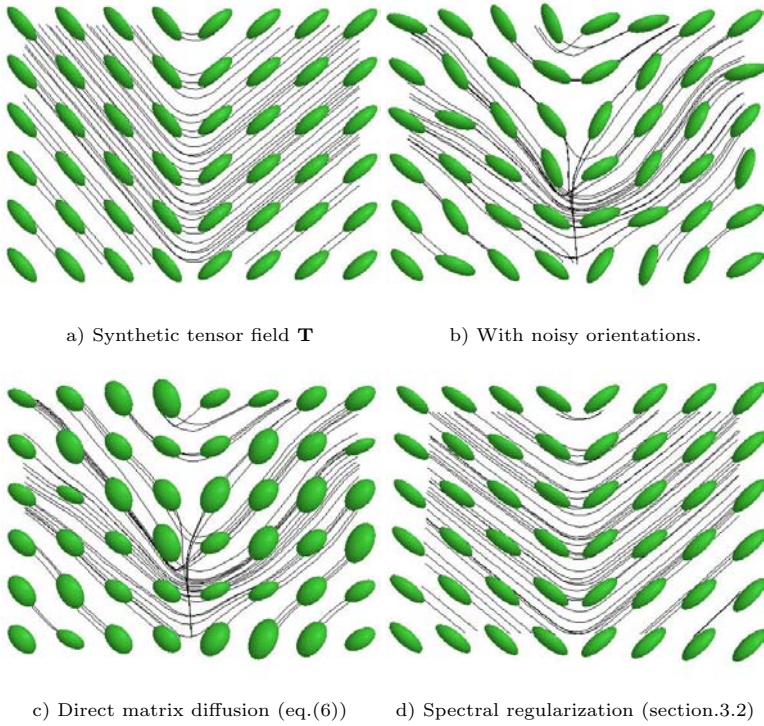


Fig. 3. Spectral versus Non-Spectral regularization methods.

(three eigenvalues, $l = 1..3$) and *orientations* $\mathbf{u}^{[l]}$ (three eigenvectors). Actually, the tensors are decomposed into :

$$\mathbf{T} = \mathbf{U} \mathbf{\Gamma} \mathbf{U}^T \quad \text{where } \mathbf{\Gamma} = \text{diag}(\lambda_1, \lambda_2, \lambda_3) \quad \text{and} \quad \mathbf{U} = \left(\mathbf{u}^{[1]} \mid \mathbf{u}^{[2]} \mid \mathbf{u}^{[3]} \right)$$

This is for instance the matter of the papers [9,11,24]. Indeed, the undesired eigenvalues swelling effect can be avoided by regularizing tensor eigenvalues more slowly than tensor orientations. The smoothing process must also consider the tensor constraints (positivity, symmetry) *in the spectral space*, which are expressed as :

$$\begin{cases} \text{Positivity : } \forall l & \lambda_l \geq 0 \\ \text{Symmetry : } \forall k, l, & \mathbf{u}^{[k]} \cdot \mathbf{u}^{[l]} = \delta_{k,l} \end{cases} \quad (\mathbf{U} \text{ is an orthogonal matrix}) \quad (7)$$

Different methods have been already propose to regularize these two spectral fields :

- **Regularization of the tensor diffusivities** : Tensor diffusivities are considered as a multi-channel image, with 3 components $(\lambda_1, \lambda_2, \lambda_3)$ and can

then be regularized with anisotropic PDE schemes, already proposed in the literature for this kind of image [13,20,22,23,25,26]). Moreover, one can easily drive the diffusivities regularization by considering specific DT-MRI indices, like mean diffusivity, fractional anisotropy, etc. (Fig.4).

$$\frac{\partial \lambda_l}{\partial t} = \text{div} (\mathbf{D}(\lambda_l, FA, VR, \dots) \nabla \lambda_l) \quad (8)$$

The positivity constraint of theses eigenvalues λ_l is simply ensured by using a scheme that satisfies the *maximum and minimum principle* [2].

- **Regularization of the tensor orientations** : The difficult part of the spectral regularization methods come from the regularization of the tensor orientations. In [11,24], the authors propose to regularize only the field of the principal direction $\mathbf{u}^{[1]}$, using a modified version of the norm constrained TV-regularization, as defined in [7]. Then, the two other tensor directions $\mathbf{u}^{[2]}$ and $\mathbf{u}^{[3]}$ are rebuild from the original noisy tensor orientation $\mathbf{U}$ and the regularized principal direction $\tilde{\mathbf{u}}^{[1]}$.

In [24], we proposed to process directly the orientation matrix $\mathbf{U}$ with a specific *orthogonal matrix-preserving* PDE flow, that anisotropically regularized the field :

$$\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{L} + \mathbf{U} \mathbf{L}^T \mathbf{U}$$

where $\mathbf{L}$ is the matrix corresponding to the *unconstrained regularization term*.

- **The orientation swapping problem** : However, when dealing with diffusion tensors, one has to take care of the non-uniqueness of the spectral decomposition $\mathbf{T} = \sum_{k=1}^n \lambda_k \mathbf{u}^{[k]} \mathbf{u}^{[k]T}$. Flipping one eigenvector direction while keeping its orientation (i.e considering $-\mathbf{u}^{[l]}$ instead of $\mathbf{u}^{[l]}$) gives the same tensor $\mathbf{T}$. It means that a constant tensor field may be decomposed into *highly discontinuous* orientation fields $\mathbf{U}$, disturbing the anisotropic regularization process with false discontinuity detections.

To overcome this problem, authors of [11,24] proposed a *local eigenvector alignment process* that is done before applying the PDE on each tensor of the field $\mathbf{T}$. However, this is a very time-consuming process which dramatically slows down the algorithms.

3.3 Isospectral Flow and Orientation Regularization

An alternative method exists, avoiding any eigenvector realignment problems. The idea lies on the use of an *isospectral flow*, that regularizes the tensor field *while preserving the eigenvalues of the considered tensors*. As a result, only tensor orientations are regularized. As we measure directly the tensor field variations from the gradients of the matrix coefficients, no false discontinuities are considered. The general form of an isospectral matrix flow is (see [9,10]) :

$$\frac{\partial \mathbf{T}}{\partial t} = [\mathbf{T}, [\mathbf{T}, -(\mathbf{G} + \mathbf{G}^T)]] \quad \text{with} \quad [\mathbf{A}, \mathbf{X}] = \mathbf{A}\mathbf{X} - \mathbf{X}\mathbf{A} \quad (9)$$

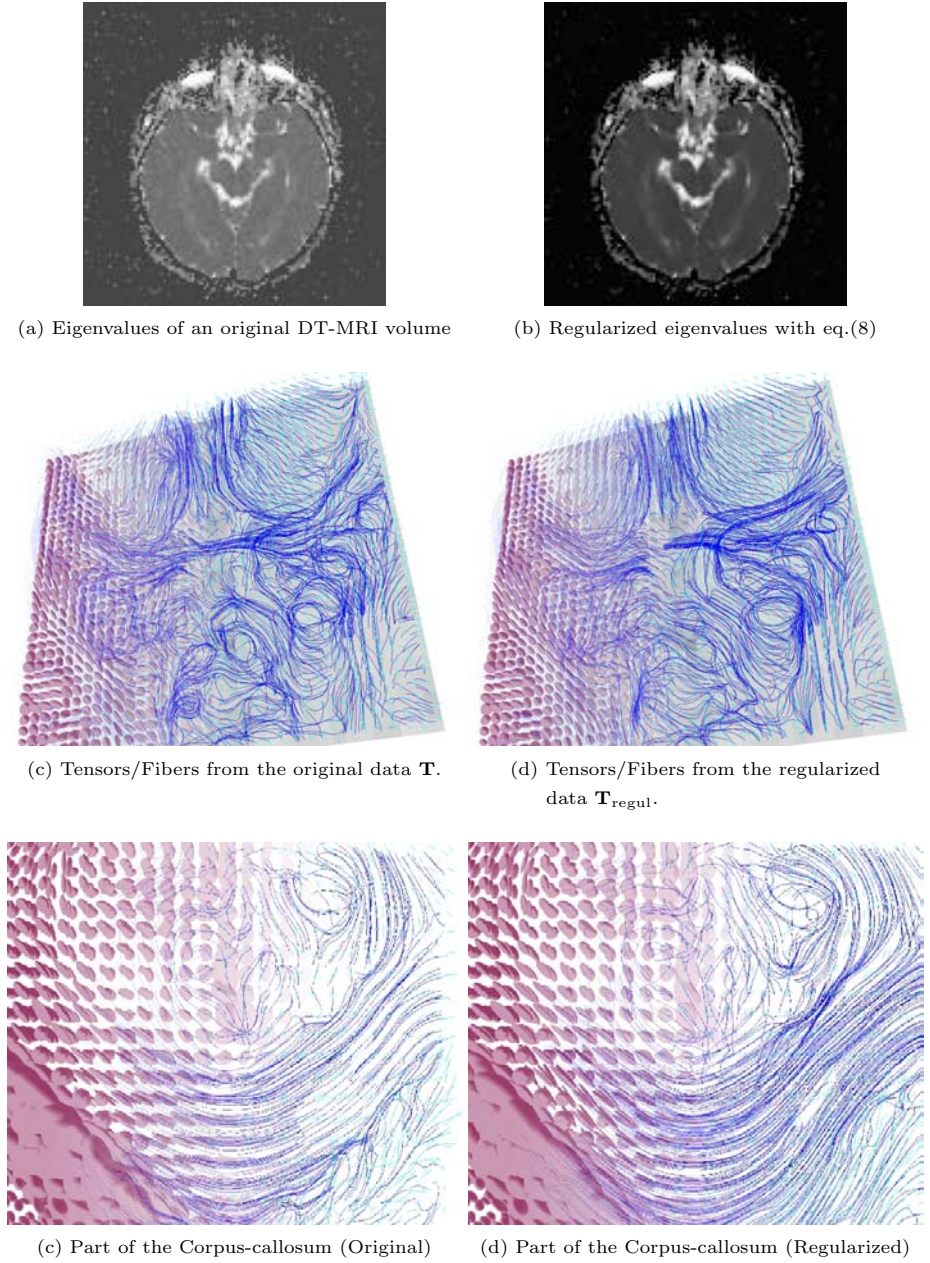


Fig. 4. DT-MRI Regularization, using constrained spectral methods (eq.(8) and eq.(9)).

Here, we choose the term $\mathbf{G}$ to correspond to the unconstrained form of the desired regularization process. It can be freely chosen. For instance, we used :

$$\mathbf{G} = (G_{i,j}) \quad \text{with} \quad G_{i,j} = \operatorname{div} \left(\frac{\phi'(\|\nabla \mathbf{T}\|)}{\|\nabla \mathbf{T}\|} \nabla T_{i,j} \right)$$

where $\phi(s) = \sqrt{1 + s^2}$ is a classical ϕ -function leading to discontinuity-preserving regularization [8]. Note that other regularization terms $\mathbf{G}$ may be suitable, as those proposed in [13,20,23,25,28], since the Eq.(9) is a very general formalism to work on diffusion tensor orientations.

Like the estimation method, a specific numerical scheme based on matrix exponentials can be used to implement the isospectral PDE flow (9), avoiding any problems of numerical rejections (see [9] for details) :

$$\mathbf{T}_{(t+dt)} = \mathbf{A}^T \mathbf{T}_{(t)} \mathbf{A} \quad \text{with} \quad \mathbf{A} = \exp(-dt[\mathbf{G} + \mathbf{G}^T, \mathbf{T}_{(t)}])$$

This equation allows to speed up the process, since no eigenvector alignment is no more necessary. Moreover, the genericity of this approach allows to combine precise and adapted regularization terms with the advantage of the separate regularization of tensor orientations and diffusivities. The exponential maps-based scheme is numerically computed using *Padé approximations* [12] for matrix exponentials, while the unconstrained regularization term $\mathbf{G}$ is discretized with classical finite differences schemes.

4 Application to Real DT-MRI Datasets

We applied our proposed isospectral-based regularization algorithm in order to improve the fiber tracking on a real DT-MRI dataset (consisting in 121 images $128 \times 128 \times 56$, courtesy of CEA-SHFJ/Orsay-France). We first estimated the diffusion tensor field from the raw images, using our robust tensor estimation method eq.(5). Then, we regularized this obtained volume of tensors with our proposed spectral methods (eq.(8) and eq.(9)) (illustration on Fig.4).

5 Conclusion and Perspectives

We proposed original PDE-based alternatives to classical algorithms used to solve two crucial problems encountered in DT-MRI imaging. Our estimation and regularization algorithms ensures the positive-definite constraint of the tensors, thanks to specific constrained variational flows and corresponding numerical schemes based on the use of exponential maps. It leads then to fast and numerically stable algorithms. Finally, we illustrate these algorithms with fiber tractography in the white matter of the brain. As a perspective, we are working on similar constrained variational methods for more coherent fiber tracking, as proposed in [4,6,16,17,19,27].

Acknowledgments. The authors would like to thank J.F Mangin (SHFJ-CEA), O.Faugeras, T. Papadopoulo (Odysée/INRIA) for providing us with the data and for fruitful discussions.

References

1. L. Alvarez, P.L. Lions, and J.M. Morel. Image selective smoothing and edge detection by nonlinear diffusion (II). *SIAM Journal of Numerical Analysis*, 29:845–866, 1992.
2. Bart M. ter Haar Romeny. *Geometry-driven diffusion in computer vision*. Computational imaging and vision. Kluwer Academic Publishers, 1994.
3. P.J. Basser, J. Mattiello, and D. LeBihan. Estimation of the effective self-diffusion tensor from the nmr spin echo. *Journal of Magnetic Resonance*, B(103):247–254, 1994.
4. P.J. Basser, S. PAJEVIC, C. Pierpaoli, and A. Aldroubi. Fiber-tract following in the human brain using dt-mri data. *IEICE TRANS. INF & SYST.*, E(85):15–21, January 2002.
5. T. Brox and J. Weickert. Nonlinear matrix diffusion for optic flow estimation. In *DAGM-Symposium*, pages 446–453, 2002.
6. J.S.W. Campbell, K. Siddiqi, B.C. Vemuri, and G.B Pike. A geometric flow for white matter fibre tract reconstruction. In *IEEE International Symposium on Biomedical Imaging Conference Proceedings*, pages 505–508, July 2002.
7. T. Chan and J. Shen. Variational restoration of non-flat image features : Models and algorithms. *Research Report. Computational and applied mathematics department of mathematics Los Angeles.*, June 1999.
8. P. Charbonnier, G. Aubert, M. Blanc-Féraud, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of the International Conference on Image Processing*, volume II, pages 168–172, 1994.
9. C. Ched'hotel, D. Tschumperlé, R. Deriche, and O. Faugeras. Constrained flows on matrix-valued functions : application to diffusion tensor regularization. In *Proceedings of ECCV'02*, June 2002.
10. M.T. Chu. A list of matrix flows with applications. Technical report, Department of Mathematics, North Carolina State University, 1990.
11. O. Coulon, D.C. Alexander, and S.R. Arridge. A geometrical approach to 3d diffusion tensor magnetic resonance image regularisation. Technical report, Department of Computer Science, University College London., 2001.
12. Gene H. Golub and Charles F. Van Loan. *Matrix computations*. The John Hopkins University Press, Baltimore, Maryland, 1983.
13. R. Kimmel, R. Malladi, and N. Sochen. Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images. *International Journal of Computer Vision*, 39(2):111–129, September 2000.
14. D. Le Bihan. Methods and applications of diffusion mri. In I.R. Young, editor, *Magnetic Resonance Imaging and Spectroscopy in Medicine and Biology*. John Wiley and Sons, 2000.
15. H. Mamata, Y. Mamata, C.F. Westin, M.E. Shenton, F.A. Jolesz, and S.E. Maier. High-resolution line-scan diffusion-tensor mri of white matter fiber tract anatomy. In *In AJNR Am NeuroRadiology*, volume 23, pages 67–75, 2002.
16. S. Mori, B.J. Crain, V.P. Chacko, and P.C.M. Van Zijl. Three-dimensional tracking of axonal projections in the brain by magnetic resonance imaging. *Annals of Neurology*, 45(2):265–269, February 1999.
17. G.J.M Parker, C.A.M Wheeler-Kingshott, and G.J. Barker. Distributed anatomical brain connectivity derived from diffusion tensor imaging. In *IPMI*, number LNCS2082, pages 106–120, 2001.

18. C. Poupon. *Détection des faisceaux de fibres de la substance blanche pour l'étude de la connectivité anatomique cérébrale*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, December 1999.
19. C. Poupon, J.F Mangin, V. Frouin, J. Regis, F. Poupon, M. Pachot-Clouard, D. Le Bihan, and I. Bloch. Regularization of mr diffusion tensor maps for tracking brain white matter bundles. In W.M. Wells, A. Colchester, and S. Delp, editors, *Medical Image Computing and Computer-Assisted Intervention-MICCAI'98*, number 1496 in Lecture Notes in Computer Science, pages 489–498, Cambridge, MA, USA, October 1998. Springer.
20. G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, 2001.
21. E.O. Stejskal and J.E. Tanner. Spin diffusion measurements: spin echoes in the presence of a time-dependent field gradient. *Journal of Chemical Physics*, 42:288–292, 1965.
22. B. Tang, G. Sapiro, and V. Caselles. Diffusion of general data on non-flat manifolds via harmonic maps theory : The direction diffusion case. *The International Journal of Computer Vision*, 36(2):149–161, February 2000.
23. D. Tschumperlé. *PDE's Based Regularization of Multivalued Images and Applications*. PhD thesis, Université de Nice-Sophia Antipolis, December 2002.
24. D. Tschumperlé and R. Deriche. Diffusion tensor regularization with constraints preservation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai Marriott, Hawaii, December 2001.
25. D. Tschumperlé and R. Deriche. Diffusion PDE's on Vector-Valued images. *IEEE Signal Processing Magazine*, 19(5):16–25, 2002.
26. D. Tschumperlé and R. Deriche. Vector-valued image regularization with PDE's : A common framework for different applications. In *IEEE Conference on Computer Vision and Pattern Recognition*, Madison, Wisconsin (United States), June 2003.
27. B. Vemuri, Y. Chen, M. Rao, T. McGraw, T. Mareci, and Z. Wang. Fiber tract mapping from diffusion tensor mri. In *1st IEEE Workshop on Variational and Level Set Methods in Computer Vision (VLSM'01)*, July 2001.
28. J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner-Verlag, Stuttgart, 1998.
29. C.F. Westin and S.E. Maier. A dual tensor basis solution to the stejskal-tanner equations for dt-mri. In *Proceedings of International Society for Magnetic Resonance in Medicine*, 2002.
30. C.F Westin, S.E Maier, H. Mamata, A. Nabavi, F.A. Jolesz, and R. Kikinis. Processing and visualization for diffusion tensor mri. In *In proceedings of Medical Image Analysis*, volume 6, pages 93–108, 2002.

An Efficient Algorithm for Multiple Sclerosis Lesion Segmentation from Brain MRI

Rubén Cárdenes^{1,3}, Simon K. Warfield², Elsa M. Macías¹, Jose Aurelio Santana³, and Juan Ruiz-Alzola^{3,2}

¹ Dept. Ingeniería Telemática,
University of Las Palmas de Gran Canaria, SPAIN
`elsa@dit.ulpgc.es`

² Harvard Medical School and
Brigham & Women's Hospital,
Dept. Radiology, USA
`warfield@bwh.harvard.edu`

³ Medical Technology Center,
University of Las Palmas de Gran Canaria and
Gran Canaria Dr. Negrín Hospital, SPAIN
`{ruben,jaurelio,jruiz}@ctm.ulpgc.es`

Abstract. We propose a novel method for the segmentation of Multiple Sclerosis (MS) lesions in MRI. The method is based on a three-step approach: first a conventional k -NN classifier is applied to pre-classify gray matter (GM), white matter (WM), cerebro-spinal fluid (CSF) and MS lesions from a set of prototypes selected by an expert. Second, the classification of problematic patterns is resolved computing a fast distance transformation (DT) algorithm from the set of prototypes in the Euclidean space defined by the MRI dataset. Finally, a connected component filtering algorithm is used to remove lesion voxels not connected to the real lesions. This method uses distance information together with intensity information to improve the accuracy of lesion segmentation and, thus, it is specially useful when MS lesions have similar intensity values than other tissues. It is also well suited for interactive segmentations due to its efficiency. Results are shown on real MRI data as well as on a standard database of synthetic images.

1 Introduction

MS lesions have similar MRI intensity levels than other tissues, such as gray matter (GM) in T1 weighted MRI or cerebrospinal fluid (CSF) in T2 weighted MRI, making lesion segmentation and subsequent quantitative analysis a very difficult task. Approaches to tackle this problem include multimodal acquisition of different datasets and atlas registration, in order to take advantage of the fact that lesions only appear inside the white matter [1].

Here we propose a supervised segmentation method based on three steps: (1) k -Nearest Neighbor (k -NN) classification, (2) distance-based intensity overlap resolution, and (3) connected components relabeling.

The k -NN rule [2] is a popular supervised classification method with asymptotic optimum properties, which has been used for years for MRI segmentation due to its good stability conditions [3]. The main disadvantage of this technique is its low execution performance due to the neighbor search, specially in the multimodal case. The simplest approach, the brute force method, computes for every test pattern, the distances to all the training prototypes and then chooses the k nearest ones. This approach is unfeasible for large 3D medical datasets. For this reason several techniques to improve the k -NN performance have been proposed. For example Friedmann [4] ordered the prototypes by increasing distance in every channel and then, the search is reduced to the channel with less local sparsity, reducing the dimensionality of the searching space. Another method is described by Jian-Zhang [5] and Fukunaga [6] which use a branch and bound strategy to find out the nearest neighbors. A very efficient method was proposed by [7], where a fast distance transformation is used to generate a look-up table with the k nearest neighbors directly available. Our k -NN implementation is based on an improved version of this method described in [8].

In order to distinguish different tissues with overlapping intensities, spatial information is added to the method by computing a distance transformation (DT) from the training prototypes. Finally a connected component algorithm is used to remove isolated voxels that are not connected to the real MS lesion.

Indicative execution times are: 0.5 seconds for 256×256 images, and one minute for a full 3D ($181 \times 181 \times 217$) volume.

2 Method

In this work we have chosen a supervised segmentation method, such as the k -NN rule, mainly because it can take into account abnormal anatomy better than unsupervised methods. Some authors have reported unsupervised methods in order to segment abnormal anatomy, as for example tumors detection [9]. Those methods need the introduction of a priori anatomy knowledge in order to detect outliers, and this is achieved usually by means of atlas registration. There are basically two problems with unsupervised methods. The first one is the accuracy, because image registration is an ill posed problem, and thus it is very hard to find voxels belonging to abnormal anatomy among the voxels that are not well registered with an atlas. Usually even for a medical expert it is hard to identify where exactly is located a lesion. The second drawback is the low speed-performance of unsupervised segmentation algorithms, which is also a clear disadvantage.

The k -NN rule is a non-parametric pattern classification technique that has proved to be an excellent method for supervised classification of MRI data. An excellent description of k -NN classification is provided in [10]. Before being applied it requires a training step, carried out by a medical expert by selecting a dataset that consists of a set of classified voxels (training prototypes). Training is carried out in 2 – 3 minutes using 100 – 300 prototypes, which are usually stored

in a file that can be interactively edited in order to improve the segmentation. After training, the method consists of a pipeline with three automatic steps:

2.1 k -NN Pre-segmentation

The k nearest prototypes of every voxel are searched in order to classify it as class c if most of the k prototypes found belong to class c . A brute force implementation of this algorithm demands a huge computational load and is not feasible for large 3D medical datasets. Thus, we have used a scheme, first described by Warfield [7], where a look-up table is computed by finding out a partition of the feature space into regions such that points in each region have the same k closest prototypes. This partition of the space is denoted as a Voronoi diagram of order k , and if the voxel values are the identifiers of the Voronoi cell centers, this is called the nearest neighbor transform of order k . Some good surveys of the types of Voronoi diagrams and its properties can be found in Okabe et al. [11] and Aurenhammer et al. [12].

In order to compute the k -Voronoi diagrams efficiently we use a novel implementation of a fast algorithm initially proposed by Cuisenaire [8], which is based on a DT by ordered propagation. The concept of ordered propagation to compute DT was introduced by Verwer [13], and the idea is essentially to scan the data from the training prototypes to the rest of the image, i.e., by increasing distance order. For this reason, every voxel in the volume is visited only once to compute the DT as opposed to distance transformations based on mask propagation [14,15,16], which need several raster scans over the image, and hence voxels are visited more than once. The difference between our approach and that of Cuisenaire is that he uses a technique called *bucket sorting* to store the elements in the propagation scheme, and our implementation uses a *double list* strategy which is less memory consuming and therefore more efficient. The k -Voronoi diagram implementation that we use requires that every pattern is visited k times, obtaining a computational complexity of order $O(k.m)$, i.e. the performance of this approach is linear in the number of patterns in the feature space, m , and linear in k .

Figure 1(a) shows the training prototypes corresponding to a two-channel MRI dataset, (T1w and T2w). The DT from the prototypes, the look-up table for $k = 1$ used in the k -NN classification, and the Voronoi diagram of order one from the prototypes are also shown in 1(b), 1(c) and 1(d) respectively. The lookup table can be interpreted as a partition of the feature space on N_c regions, where N_c is the number of classes, and each region represents a class c , in such a way that a pattern is classified as class c_i if its feature coordinates are located in the region of class c_i at the lookup table. Notice that the lookup table is formed by the junction of the Voronoi cells belonging to the same class. We show the result of the pre-segmentation step with one channel for a T1w MRI coronal slice of the brainweb dataset [17] in figure 3(b), where many voxels belonging to gray matter are misclassified as MS and vice versa due to the overlapping in the intensity space.

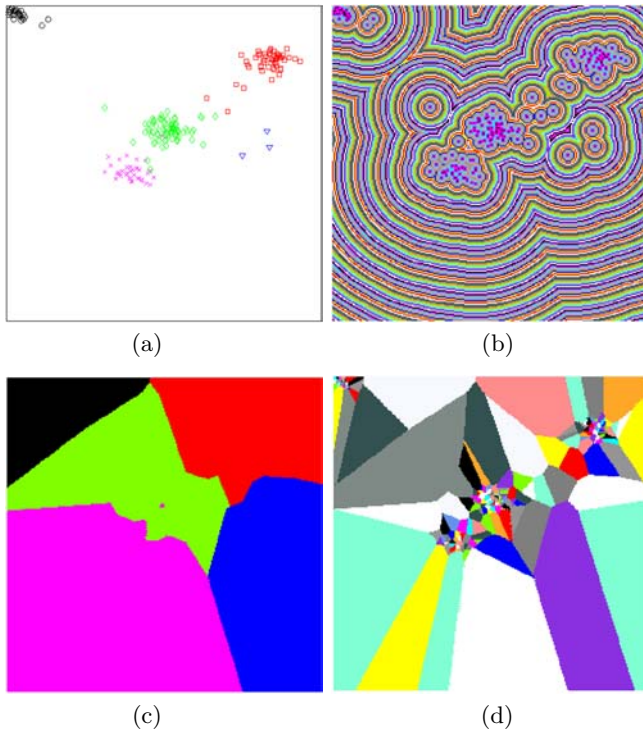


Fig. 1. Training prototype patterns corresponding to a two channel MRI data (a), distance transformation coded with a cyclic colormap from the prototypes (b), lookup table for $k = 1$ (c) and order one Voronoi diagram corresponding to the training prototypes (d)

2.2 Distance-Based Relabeling

The initial k -NN classification using only intensity information is not enough to get good results if there exist tissues with intensity values in the same intensity range, as for example gray matter and lesion in T1 weighted MRI. For this reason we propose to separate these ambiguous tissues using additional information. This information is taken here from the spatial location of the training prototypes, in particular, the Euclidean distance from a given pattern to the nearest prototype in the image space. With the addition of spatial information, the patterns to classify becomes of the form $(I_1, \dots, I_D, dx, dy, dz)$, increasing in 3 the space dimension: $D+3$, where D is the number of channels used. This means that the neighbor search is now in a space of dimension 4 or 5 if the number of channels used are 1 or 2. Searching in a space of dimension higher than 3 is more computationally expensive, more memory consuming and harder to implement. Thus, focusing on efficiency, our method separates intensity and spatial information in different steps. We propose to use a distance-based relabeling step

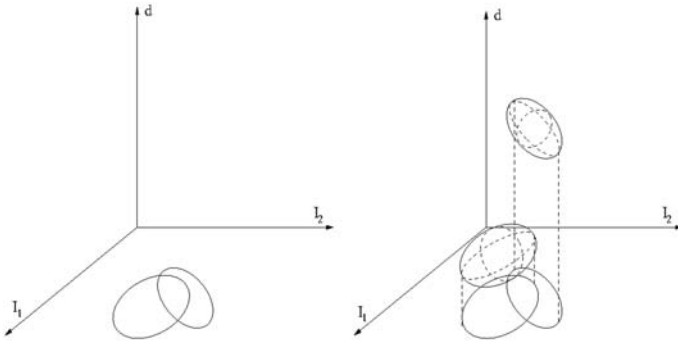


Fig. 2. Two classes overlapped in the feature space I_1 and I_2 (left), and class separation adding the spatial channel d (right)

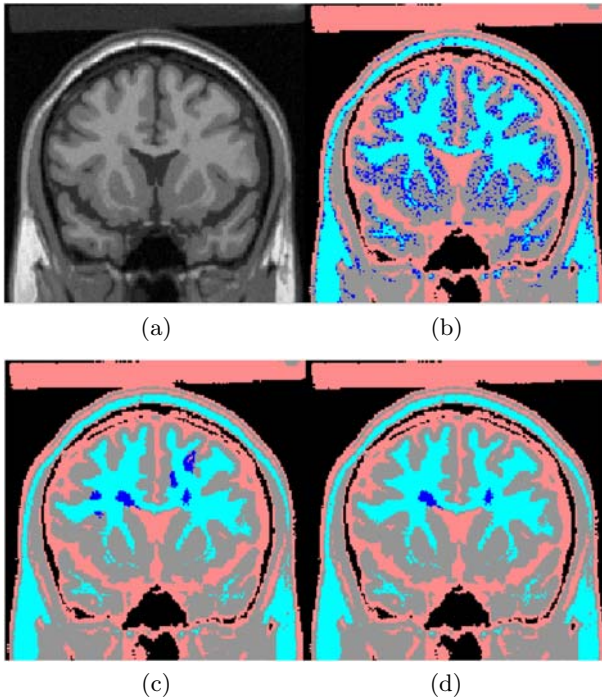


Fig. 3. Source image: T1w MR coronal slice (a). First step: k -NN segmentation with one intensity channel (b). Second step: distance-based relabeling (c). Third step: connected component filtering from MS lesions (d). Color code: Background = black, White Matter = light blue, Gray Matter = gray, CSF = orange, MS = dark blue

in addition to the previous k -NN pre-segmentation, in order to add the spatial information and distinguish patterns with similar intensity values that belong to different classes.

Let C be the set of classes with overlapping intensity values. If a pattern p is classified as class $c_i \in C$, p will be relabeled, assigning to it the class of the prototype closest to it. This is like adding a new channel of distances d to the k -NN classification in order to resolve the overlap of the classes in the intensity space, as shown schematically in figure 2.

In order to make the relabeling of the pre-classified voxels in an efficient way, the Euclidean distance transformation from the prototypes belonging to the set of overlapping classes C are computed in the whole image. For example, in a T1 weighted MRI, there are two conflicting classes in C : gray matter (c_1) and lesion (c_2). In this case two DTs are computed from the prototypes of both classes. Then a pattern p initially belonging to c_1 , will be classified as c_2 if the DT from c_2 prototypes has a lower value at p than the DT from c_1 prototypes. Notice that we prevent mis-classifications of voxels clearly classified, regardless how close in the image are the prototypes of the other classes, by avoiding relabeling voxels outside C , i.e. voxels with no intensity overlapping.

As in the first step, the DT implementation used here is based on ordered propagation with a *double list* strategy in order to be highly efficient. In consequence, the computational complexity in this step is linear in the number of voxels: $O(M)$. Notice that now we need to compute a DT in a 3D space, while in the previous k -NN pre-classification step the lookup table was computed in a domain of dimension equal to the number of channels used, which is usually no more than two. Therefore, the computational load is now higher because M is in general greater than $k.m$. The result of the distance-based relabeling step for the T1w MRI coronal slice of the brainweb dataset is shown in figure 3(c), and the DT used is shown in figure 4 for the GM prototypes and the MS prototypes.

2.3 Connected Component Filtering

Still some mis-classifications remain since there are voxels located near lesion prototypes but they do not really belong to lesion, as shown in figure 3(c). This problem appears because the proportion of lesion prototypes respect to the number of voxels that really belong to lesion is high compared to this proportion in other tissues due to the major importance and locality of MS lesions. For this reason many non lesion voxels are mis-classified. The third step corrects this problem, removing the lesion regions not connected to lesion prototypes. This is achieved applying a connected component filter to the classified image starting from the lesion prototypes (see figure 3(d)). It is required to have at least one prototype for every MS lesion connected component, in order to obtain a correct classification with our method.

2.4 Results

In addition to the 2D experiment shown before, we have made 3D experiments using the standard simulated $181 \times 181 \times 217$ MRI brainweb dataset [17] to asses quantitatively the speed-performance, we illustrate in figure 6 the orthogonal slices of a 3D segmentation with our method. The total CPU times measured

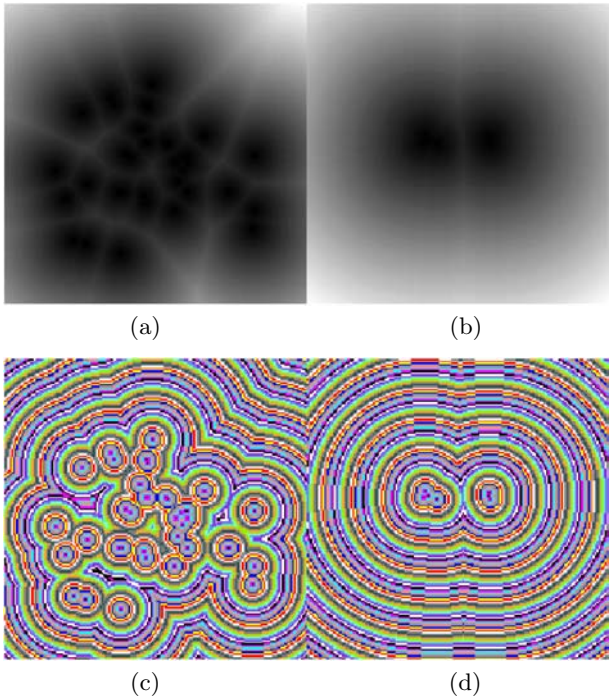


Fig. 4. DT from GM prototypes (a),(c) and from MS prototypes (b),(d) coded in gray scale colormap (top), and coded in a cyclic colormap (bottom)

in a SUN Ultra-10 workstation with a SPARC II 440 MHz processor and 512 MB RAM, are shown in table 1, as well as the partial times for the different steps in the 2D and 3D experiments. Notice that the second step is the most time consuming one, and the total execution time in the 3D experiment takes less than one minute (59.66 seconds).

Figure 5 shows an example from real MRI data, from a patient with Multiple Sclerosis. The original axial slices PD and T2, show four non connected MS lesions, which are always correctly identified by our method using $k = 5$ and 100 prototypes. Notice that if no distance information is used, multiple MS misclassifications appear in the image, as shown in figure 5 on the left side of the last three rows, moreover when the intensity values are similar to other structures and only one channel is used as in the T2 case shown in the second row of figure 5.

Table 1. Total and partial execution times in seconds, for the 2D and 3D experiments. Number of prototypes used: 127 in the 2D case, and 390 in the 3D case

	step 1	step 2	step 3	Total
3D (181x181x217)	6.67	52.36	0.63	59.66
2D (181x181)	0.09	0.26	0.002	0.35

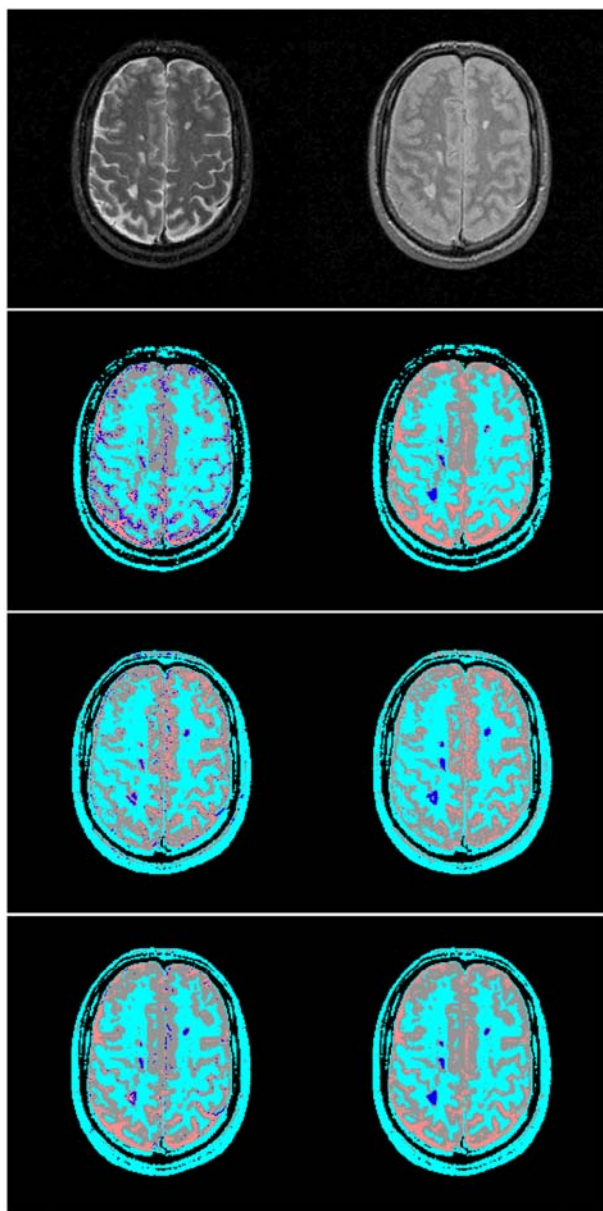


Fig. 5. Top row: axial T2 weighted MRI slice of a patient with MS, (left) and PD weighted (right). Second row: k -NN segmentation using T2 (left) and using T2 plus spatial information (right). Third row: k -NN segmentation using PD (left) and using PD plus spatial information (right). Bottom row: k -NN segmentation using T2 and PD (left) and using PD, T2 plus spatial information (right). Color code: Background = black, White Matter = light blue, Gray Matter = gray, CSF= orange, MS = dark blue

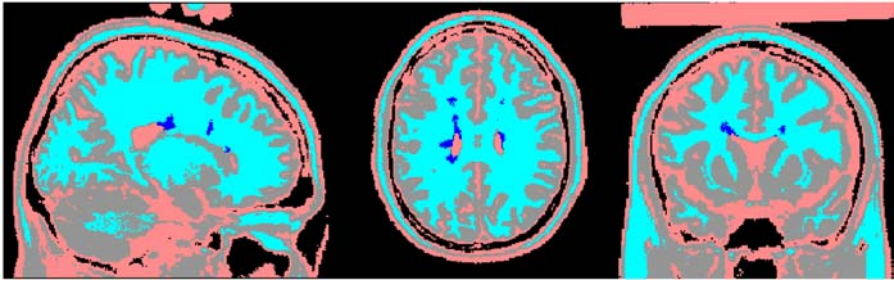


Fig. 6. Orthogonal slices of a 3D segmentation

3 Conclusions

We have shown a novel high performance method for the segmentation of abnormal anatomy in MRI data, such as MS lesions. One of the main features of this scheme is that it can segment different structures with the same intensity level range. The other principal feature is the high performance achieved due to the fast algorithms to compute distance transformations and Voronoi diagrams on which our method is based on.

Our scheme also shows some advantages with respect to unsupervised methods, because it is fairly stable for the segmentation of abnormal anatomy, and because no image-atlas registration is needed, which is usually a performance bottleneck in other methods. On the other hand, the whole execution time is to be increased around one more minute in order take into account the user interaction to train the classifier.

The algorithm shows a high accuracy, depending essentially on the training dataset selected by a medical expert, and it performs really well using one intensity channel compared to segmentations carried out with more than one channel, which is a clear advantage for clinical applications. It is useful for interactive segmentation due to its high performance and the facility to add or remove training prototypes to improve the results.

The applications of this method go well beyond MS MRI segmentation since it can be used to segment almost every type of image modalities. Currently we are starting to use it for MRI segmentation of the knee cartilage.

Acknowledgments. The first author is funded by a FPU grant from the University of Las Palmas de Gran Canaria. This work has been partially supported by the Spanish Ministry of Science and Technology and European Commission, co-funded grant TIC-2001-38008-C02-01.

References

1. Warfield, S., Kaus, M., Jolesz, F., Kikinis, R.: Adaptive, template moderated, spatially varying statistical classification. *Medical Image Analysis* **4** (2000) 43–55

2. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Transactions on Information Theory* **IT-13(1)** (1967) 21–27
3. Clarke, L., Velthuizen, R., Phuphanich, S., Schellenberg, J., Arrington, J., Silbiger, M.: MRI: Stability of three supervised segmentation techniques. *Magnetic Resonance Imaging* **11** (1993) 95–106
4. Friedman, J., Baskett, F., Shustek, L.: An algorithm for finding nearest neighbors. *IEEE Trans. on Computers* **C-24(10)** (1975) 1000–1006
5. Jian, Q., Zhang, W.: An improved method for finding nearest neighbors. *Pattern Recognition Letters* **14** (1993) 531–535
6. Fukunaga, K., Narendra, P.: A branch and bound algorithm for computing k-nearest neighbors. *IEEE Transactions On Computers* **C-24** (1975) 750–753
7. Warfield, S.: Fast k-nn classification for multichannel image data. *Pattern Recognition Letters* **17(7)** (1996) 713–721
8. Cuisenaire, O., Macq, B.: Fast k-nn classification with an optimal k-distance transformation algorithm. *Proc. 10th European Signal Processing Conf.* (2000) 1365–1368
9. Kaus, M.: Contributions to the Automated Segmentation of Brain Tumors in MRI. PhD thesis, Berlin, Germany (2000)
10. Duda, R., Hart, P.: *Pattern Classification and Scene Analysis*. John Wiley & Sons (1973)
11. Okabe, A., Boots, B., Sugihara, K.: *Spatial Tesselations: Concepts and Applications of Voronoi Diagrams*. Wiley (1992)
12. Aurenhammer, F., Klein, R.: Voronoi diagrams. In Sack, J., Urrutia, G., eds.: *Handbook of Computational Geometry*. Elsevier Science Publishing (2000) 201–290
13. Verwer, B., Verbeek, P., Dekker, S.: An efficient uniform cost algorithm applied to distance transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11(4)** (1989) 425–429
14. Borgefors, G.: Distance transformations in arbitrary dimensions. *Computer Vision, Graphics and Image Processing* **27** (1984) 321–345
15. Danielsson, P.: Euclidean distance mapping. *Comput. Graph. Image Process.* **14** (1980) 227–248
16. Ragnemalm, I.: The Euclidean distance transform in arbitrary dimensions. *Pattern Recognition Letters* **14** (1993) 883–888
17. Cocosco, C., Kollokian, V., Kwan, R.S., Evans, A.: Brainweb: online interface to a 3D MRI simulated brain database. In: *Neuroimage*. Volume 5 of 425., Copenhagen (1997)

Dynamical Components Analysis of fMRI Data: A Second Order Solution

Bertrand Thirion and Olivier Faugeras

Odyssée Laboratory (ENPC-Cermics/ENS-Ulm/INRIA)

Abstract. In this paper, we present a new way to derive low dimensional representations of functional MRI datasets. This is done by introducing a state-space formalism, where the state corresponds to the components whose dynamical structure is of interest. The rank of the selected state space is chosen by statistical comparison with null datasets. We study the validity of our estimation scheme on a synthetic dataset, and show on a real dataset how the interpretation of the complex dynamics of fMRI data is facilitated by the use of low-dimensional, denoised representations. This methods makes a minimal use of priors on the data structure, so that it is very practical for exploratory data analysis.

1 Introduction

Functional Magnetic Resonance Imaging (fMRI) of blood oxygen level-dependent (BOLD) contrast has become a common tool for the localization of brain processes associated with all kinds of psychological tasks. fMRI data analysis has first been performed from a hypothesis testing point of view, in which the experimenter tests the presence of a predicted effect in the data. The success of this approach is largely due to the relatively intuitive and efficient methods (general linear model) proposed to analyze the data [5]. Nevertheless, the power of this kind of methods is related to the validity of the underlying hypotheses, so that there is also a need for less constrained methods that give an account of the data prior to hypothesis testing: this is what exploratory analysis deals with (see [4], for example).

To date, the most popular exploratory technique for fMRI data is probably spatial independent components analysis (ICA), which aims at disentangling the data into spatially independent components [8]. However, this method ignores the temporal content of the data, which actually allows for the interpretation of the components in terms of task-related activity. In fact, if activation signals are weak, it is not certain that any ICA algorithm can really discover them (notably after a PCA data reduction). The solution to that problem is to model the temporal content of the data together with the spatial unmixing.

In [11] we proposed a dynamical components analysis (DCA) of fMRI data that aimed at both separating spatio-temporal components and characterizing the temporal signals. The introduction of a nonlinear functional on the set of possible decompositions led to several difficulties in the optimization procedure.

Here we propose a much simpler setting which can be used to solve the problem with a limited, i.e. up to second order, statistical analysis of the data.

The remainder of this paper is organized as follows: in section 2, we present our state-space approach to solve the DCA problem. We then test the detection power of our estimation procedure on synthetic data in section 3, and confront our model to real data in section 4. The method is discussed in section 5.

2 The State Space Model

Dynamical Components Analysis. Let Y be a fMRI dataset, where each $Y(t)$ is an image which is simply written as an N -dimensional vector. N is thus the number of voxels considered in the analysis, and the length of the series is T . As stated in [11], the dynamical components analysis of Y results in a decomposition:

$$Y(t) = \sum_{k=1}^K M_k x_k(t) + W(t), \quad (1)$$

where $0 < K < \min(N, T)$, $M = (M_k)$, $k = 1..K$ is a set of images; each $W(t)$ is a N -dimensional vector and $x_k(t)$ are temporal signals.

Equation (1) means that the dataset is decomposed into meaningful spatial (the M_k 's) and temporal (the x_k 's) signals plus a noise term. The problem is to estimate K , M and $X(t) = \{x_k(t)\}$ given the data Y and some prior information about the experiment (e.g. the experimental paradigm). The K -dimensional vector $X(t)$ can be thought of as the "state" representation of the measurements $Y(t)$. The experimental paradigm will be represented as a $c_e \times T$ matrix P , where c_e is the number of experimental conditions. Each row of P is the time course of an experimental condition. Let us recall that in fMRI it is usually the case that $N \gg T$.

The State-Space Model. Let us add to equation (1) another equation that defines the evolution of the system, which is the "dynamical" part of the problem. Let A_1 and A_2 be $K \times K$ and $K \times c_e$ matrices, and $A = [A_1^T, A_2^T]^T$. We propose the following model for the data:

$$X(t+1) = A_1 X(t) + A_2 P(t) + V(t) \quad (2)$$

$$Y(t) = M X(t) + W(t) \quad (3)$$

where each K -dimensional vector $V(t)$ is an *innovation process*. We call equation (3) or (1) the mixing equation, since it interprets the observation Y as a noisy mixture of the components of X . Equation (2) is the evolution equation. This equation is a well known multivariate AR(1) model, excited by the input $P(t)$. An obvious limitation of the system (2-3) is the restriction to a first order model. One formal answer is to translate higher order evolution models into a first order one by introducing an auxiliary variable $Z(t) = (X(t), X(t+1))$ and

solving a similar system in Z . However, this increases the number of unknowns. Instead, we apply recursively the model (2) which provides a system of equation:

$$X(t+p) = A_1^p X(t) + \sum_{\tau=0}^{p-1} A_1^{p-1-\tau} A_2 P(t+\tau) + \sum_{\tau=0}^{p-1} A_1^{p-1-\tau} V(t+\tau), \quad p \geq 1. \quad (4)$$

When necessary, this approach will provide us with some constraints to estimate X , A_1 and A_2 (see section 2.2).

Variations on the Same Theme. Most of the linear models that can be found in the fMRI analysis literature can be viewed as instantiations of the system (2,3): For example, the general linear model of Friston et al. [5] is obtained by setting $X = HP$ in equation (2), where H contains some hemodynamic response model. Then the linear regression is simply the least square solution of the mixing problem (equation (3)), M being the unknown. X is thus guessed, and not estimated.

The singular values decomposition (SVD) [1], Temporal ICA [2] and Spatial ICA [8] do not deal with any evolution model, but simply state a structure for the unmixing model: respectively decorrelation of the components of M and X , temporal independence for the rows of X , spatial independence for the columns of M . In the three cases, choosing the order K of the model is a challenging task. Nevertheless, some temporal ICA -like algorithms tend to maximize the autocorrelation of the signals which is a special case of equation (2) where $A_2=0$. This is in particular the case of the CCA algorithm [4].

2.1 Second Order Solution

We recall the approach of Soatto et Chiuso [10] who addressed the problem of dynamical textures recognition in a similar framework. Their procedure, unlike the most general estimation procedures [9] -in particular, the EM learning algorithm based on Kalman recursions [6]-, has the key advantage of being non-iterative. It is based only on the hypothesis of an AR(1) system. We additionally introduce prior knowledge on the system -in our case, the paradigm P .

A Linear Estimation Scheme. The general problem formulated by equations (2) and (3) does not have a unique solution. Indeed, assuming that such a decomposition has been achieved, yielding A_1, A_2, M, V, W, X , and that J is a $K \times K$ invertible matrix, one also has the solution $JA_1J^{-1}, JA_2, MJ^{-1}, JV, W, JX$. This means that the solutions of the problem are in fact equivalent classes under this action of the group $Gl(K)$. To remove this ambiguity, we force $X^T X$ to be diagonal.

As in [10], we first derive estimates for M and X and only then for (A_1, A_2) ; W and V are residuals of the estimation procedure. Assuming a deterministic evolution ($V = 0$), we are in the situation described in figure 1. Given the

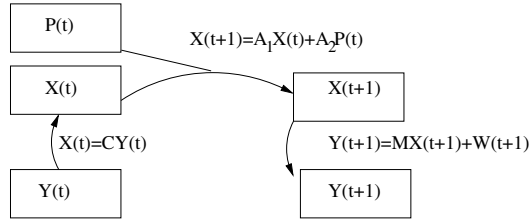


Fig. 1. Basic dynamical model in the case of a deterministic ($V = 0$) evolution.

observation noise covariance Λ_W , this model yields the following problem:

$$\min_{M,C} \|Y(t+1) - M(A_1 C Y(t) + A_2 P(t))\|_{\Lambda_W}, \quad (5)$$

where $\|Y\|_{\Lambda_W}$ stands for $Y^T \Lambda_W^{-1} Y$; C is the unmixing matrix associated with the mixing matrix M ($X(t) = C Y(t)$); for notational convenience, we extend this matrix with the identity in order to replace $A_1 C Y(t) + A_2 P(t)$ by $[A_1, A_2] C [Y(t) P(t)]$; Since Λ_W is not known, one has to approximate (5); one possibility is to make sure that the noise is white. Therefore, we first whiten the data Y in order to approximate the situation where $\Lambda_W = I_N$, the $N \times N$ identity matrix.

Let $Y_1 = [Y(1), \dots, Y(T-1)]$, $P_1 = [P(1), \dots, P(T-1)]$, $\bar{Y}_1 = [Y_1^T, P_1^T]^T$, $X_1 = [X(1), \dots, X(T-1)]$, $Y_2 = [Y(2), \dots, Y(T)]$ and $X_2 = [X(2), \dots, X(T)]$. Let L_1 and L_2 be the Cholesky decomposition of $\bar{Y}_1$ and Y_2 ($L_i L_i^T = Y_i Y_i^T$, $i = 1, 2$, and $L_i, i = 1, 2$ is trigonal). Equation (5) is thus approximated by

$$\widehat{M}, \widehat{C} = \operatorname{argmin} \|L_2^{-1} (Y_2 - M[A_1, A_2] C [Y_1, P])\|_2 \quad (6)$$

$$= \operatorname{argmin} \|Z_2 - L_2^{-1} M A C L_1 Z_1\|_2, \quad (7)$$

where we have introduced the whitened data $Z_1 = L_1^{-1} \bar{Y}_1$ and $Z_2 = L_2^{-1} Y_2$. Since we solve the problem first in terms of M and $X_1 = C Z_1$, without prior knowledge on A , we make further assumptions on A : its singular values should be less than or equal to 1, for the sake of system stability. But what makes the difference between the state and the noise is that the state of the system is temporally structured, so that the singular values of A are close to 1. Hence, we assume that A is equivalent to a K -dimensional projector; it can then be incorporated into C . Thus

$$\widehat{M}, \widehat{C} = \operatorname{argmin} \|Z_2 - L_2^{-1} M C L_1 Z_1\|_2 \quad (8)$$

$$= \operatorname{argmin} \|Z_2 Z_1^T - L_2^{-1} M C L_1\|_2 \quad (9)$$

It is a classical result that the SVD of $Z_2 Z_1^T$ provides us with the best estimate of M and $X_1 = C \bar{Y}_1$ in the least square sense. We write

$$Z_2 Z_1^T = U \Sigma \Omega^T \quad (10)$$

where U and Ω are orthogonal matrices, and Σ is diagonal. Note that by construction the singular values $\sigma_1 > \dots > \sigma_N$ are between 0 and 1 and repre-

sent the correlation between the data components at time $t + 1$ and the predictor $C\bar{Y}_1$. Under a gaussian noise hypothesis, they can equivalently be interpreted as the mutual information between these components through the formula $I_i = \frac{1}{2} \log \left(\frac{1}{1 - \sigma_i^2} \right)$. In the ideal case of noise-free mixing, $\sigma_1 > \dots > \sigma_K > 0$ and $\sigma_{K+1} = \dots = \sigma_N = 0$ (see the above hypothesis on A). In practice, one has to set a threshold. This question is addressed later. This being done, one reduces the matrices U , Σ and Ω to their first K rows, which yields U_K , Σ_K and Ω_K .

$$\widehat{M} = L_2 U_K \Sigma_K^{1/2} \quad (11)$$

$$\widehat{X}_1 = \Sigma_K^{1/2} \Omega_K^T L_1^{-1} \bar{Y}_1 = \Sigma_K^{-1/2} U_K^T L_2^{-1} Y_2 \quad (12)$$

The estimation of the mixing matrix M yields simply a least square solution for X , and thus W (in [10], a reprojection is proposed for the estimation of X_2 , but this has little impact on the final results). Given an estimate of X_1 and X_2 , we obtain $[\hat{A}_1, \hat{A}_2] = \widehat{X}_2 \cdot \text{pinv}([\widehat{X}_1^T, P_1^T])$, where *pinv* stands for the pseudo-inverse of the matrix. An estimation of V follows immediately. Note that X has indeed a diagonal covariance matrix. By contrast, A_1 is not necessarily diagonal: this formalism allows for interactions between components of the state vector X . A perhaps more intuitive way to understand this method is to interpret it in terms of projection: from equation (9), one has

$$MX_1 = MC\bar{Y}_1 = L_2 Z_2 Z_1^T L_1^{-1} \bar{Y}_1 = Y_2 Z_1^T Z_1 \quad (13)$$

since $Z_1^T Z_1 = \bar{Y}_1^T (\bar{Y}_1 \bar{Y}_1^T)^{-1} \bar{Y}_1$, $Z_1^T Z_1$ is nothing but the projector onto the rows of $\bar{Y}_1$; thus (13) simply means that MX_1 is the projection of the data at time $t + 1$ onto the input at time t . Introducing priors P in the model simply consists in *enlarging* the projection operator: one projects Y_2 onto the rows of Y_1 and P instead of Y_1 only.

2.2 Some Additional Developments

Estimation of K . We are not aware of any analytical method to determine the correct value for K , given the singular values $\sigma_1, \dots, \sigma_N$. The problem is even ill-posed: when $N > \frac{T}{2}$, $\sigma_1 = 1$ necessarily; in effect, the rows of Z_1 and Z_2 generate two subspaces of dimension N in $\mathbb{R}^T$, thus they share at least one vector. It is then impossible to test for the presence or absence of structures in the process that generated the data. We solve this issue next. Assuming -as [4]- that $N \ll T$, one estimates by simulation the distribution of σ_1 ; this involves randomly generating independent N -dimensional subspaces of $\mathbb{R}^T$ analogous to Y and computing σ_1 by equation (10). We noticed from experiments that a gaussian approximation of the distribution of σ_1 is satisfactory in practice, which allows for quick simulation procedures.

One tests the null hypothesis on the first eigenvalue σ_1 ; if the test is negative, one recursively tests the null hypothesis for σ_{k+1} , using $N - k$ dimensional subspaces of $\mathbb{R}^{T-k}$ until the null hypothesis is no longer rejected.

Generalization to Higher Orders. As we have noticed, the estimation of K breaks down when $N > \frac{T}{2}$ (and before in practice). For fMRI, $N \gg T$ always holds, but standard data reduction allows us to consider that $N \leq T$. Instead of reducing crudely N by PCA, we add some constraints in the estimation procedure. A simple idea is to use the higher order model described in equation (4); this model simply means that the prediction procedure that is used to predict $Y(t+1)$ from $Y(t)$ and $P(t)$ could also be extended to predict $Y(t+p)$ from $Y(t)$ and $P(t), \dots, P(t+p-1)$ -without making the model more complex.

In practice, the method is thus to generate the successive observation matrices $Y_1, \dots, Y_p$ (instead of Y_1 and Y_2 only), their whitened counterparts $Z_1, Z_2, \dots, Z_p$ -which may also incorporate the experimental paradigm, and then to apply recursively the projection equation (13):

$$\tilde{Z}_2 = Z_2 Z_1^T Z_1, \dots, \tilde{Z}_p = Z_p \tilde{Z}_{p-1}^T \tilde{Z}_{p-1} \quad (14)$$

The mixing matrix M is then estimated by computing the SVD of $\tilde{Z}_p$, and the estimation of X and the other quantities simply follows by least square estimation. This procedure is an iteration of the projection that defines the state space; it helps to distinguish between correlations that are embedded in the data structure and those that occur by chance. The empirical rank test described previously adapts to this generalization, by using the same iteration procedure for the surrogate data.

Given N and T , we first estimate p , i.e. how many projections will be necessary in order to enable a test for the value of K , which amounts to requiring $\sigma_1 < 1$ with a given probability. Then, one applies the recursive projection procedure to the data, including the experimental paradigm. The rank of the state process is then determined by comparison with the empirical null distribution of the eigenvalues

3 Illustration and Test on Synthetic Data

Data and Method. We simulate one slice of fMRI data containing 1963 voxels. 3 small foci of 21 voxels are created and an independent gaussian white noise is added to all voxels, so that SNR is 0.5 in the *activated areas*. The activation areas are kept small enough to have a more challenging multivariate signal extraction. The length of the series is $T = 200$; the simulated paradigm comprises two conditions (see figure 2(a)); the design is the sequence of two alternating conditions in 10 scans long blocks. The ground truth activation of the 3 foci is obtained by convolution with the canonical hemodynamic response function (hrf) of SPM, sampled at 2s, but with a different weight for the two conditions, according to each focus of activation (see figure 2(b)). The 1963 voxels data is reduced by PCA to a set of $N = 200$ time series.

Our analysis comprises two parts: First, we would like to check whether the rank of the estimated state is correct (i.e. equal to 2). Second, if the state variable contains the correct activation patterns.

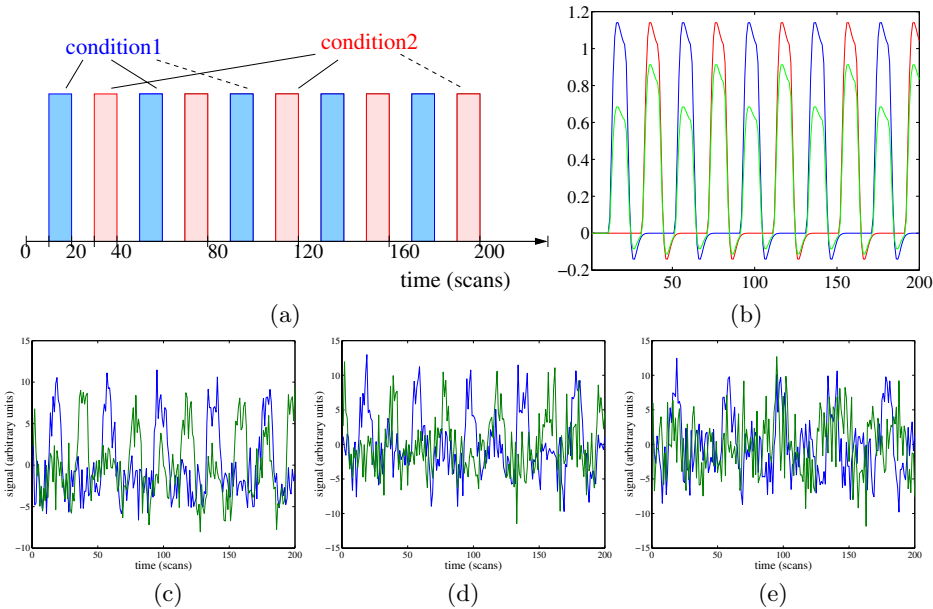


Fig. 2. (a) Simulated experimental paradigm for the synthetic experiment. The design is the succession of two alternating conditions in 10 scans long blocks, separated by 10 scans long resting periods. (b) Synthetic activations simulated in the experiment. The three activation patterns are obtained by convolution of the canonical hrf with a linear combination of the experimental conditions. The weights of the linear combination are fixed for each focus. Time courses obtained from the state-space estimation algorithm (c), a temporal CCA algorithm (d), and first two components of the PCA of the data (e).

Results. The dimension K of the state variable X has been tested for different values $N = 10, 20, \dots, 120$. The number p of recursions in the estimation of the state, as well as the results for K are given in table 1.

The dimension of the state has been estimated to 2 in all but two cases, in which it has been estimated to 1 only. This means that our rank test is correct in general, though a bit too conservative. This should be contrasted with the Bartlett test on the original data, which does not find any subspace of interest, due to the flatness of the data spectrum (not shown).

To study the results of the method, we display two vectors that span the state space as obtained for $N = 100$ in figure 2(c). They appear to be correctly defined, in the sense that they approximately span the space of the vectors displayed in

Table 1. Estimation of the rank of the state of the synthetic dataset

N	10	20	30	40	50	60	70	80	90	100	110	120
p	1	1	1	1	1	2	2	2	2	3	3	3
K	2	2	2	2	1	2	2	2	2	2	2	1

figure 2(b). Together are represented the two first components of the PCA and of the temporal CCA. It is clear that the state-space results outperform the CCA results, which in turn outperform the simple PCA. This is not surprising: The state-space model takes into account the prior information of the experimental paradigm, and comes from a higher order, hence more regular model. The CCA model, unlike PCA, yields autocorrelated patterns, that are thus more regular.

4 Application on Real Data

In this section, we study the benefit of the state-space model applied on real fMRI data. Recalling that the data is first reduced to a dimension $N < T$ by a PCA, we will be particularly concerned by the following questions:

- Is the rank estimation stable with respect to N ?
- Is there any advantage in choosing $N \sim \frac{T}{2}$ rather than $N \ll T$?
- Does the reduced state variable allow for an easy data interpretation?

4.1 Materials

Described in [13], the data belongs to a study on monkey vision: The task performed by the Rhesus monkey is the passive viewing of moving and static textures. The experimental paradigm consists in 3 repetitions of the following stimulation sequence: viewing of a static texture (random dots) during 10 scans, rest during 10 scans, viewing of a moving texture during 10 scans and rest during 10 scans, thus yielding 120 scans long sessions. The dataset considered here comprises 11 sessions. It was acquired with a 1,5T scanner. The repetition time is $TR=2.976$ seconds. One volume is $64 \times 64 \times 32$ voxels and the spatial resolution is $2 \times 2 \times 2$ mm; it comprises the whole brain. Before the experiment, the monkey had undergone an injection of MION (monocrystalline iron oxide nanoparticle) contrast agent of 4 mg/kg, so that the measured signal is not the BOLD contrast, but is related to the local cerebral blood volume [13]. Using this contrast agent is known to increase the contrast to noise ratio, but challenges the hypotheses used for standard models of fMRI data.

In this study, spatial registration was applied to the data. We further study one session, since our goal is to explore the variety of the responses to the experimental tasks. For each voxel, the time series was centered and detrended.

4.2 Results

Estimation of K . The challenge in the estimation of the rank of the system is the presence of temporal correlation in the data. Our hypothesis is that a low-dimensional signal space can account for it. K has been estimated for different values $N = 10$ to 70 -given that $T = 120$. The number p of recursions in the estimation of the state, as well as the results for K are given in table 2.

One can expect that the estimated value of K should increase with N . In fact, we rather observe that this number is stationary, indicating that probably

Table 2. Estimation of the rank of state of the real dataset

N	10	15	20	25	30	35	40	45	50	60	70
p	1	1	2	2	2	2	2	3	3	3	4
K	3	3	4	4	4	5	5	4	4	4	4

most of the autocorrelated components lie in the subspace generated by the first principal components of the data. It is not clear whether $K = 4$ or 5. Next, we choose $K = 4$.

Influence of Increasing N . It is interesting to compare the state estimations for different values of N . Since there is no obvious criterion for such a comparison, we can qualitatively study how both models approximate the signal of a voxel of interest. The time courses of a voxel $Y_n(t)$ of interest, as well as the reconstructed time courses (after reduction to $N = 20$ and 40) $M_nX(t)$ are displayed in figure 3 (a). We notice that both state spaces provide a good reconstruction model for the data, in the sense that the activation pattern is well fitted.

Description of the State-Space. It would be interesting to interpret clearly the low dimensional state space in terms of temporal effects (task-related activity, physiological effects, ill-corrected motion, trends), but we have to face the representation problem: there is an indeterminacy in the representation due to the group action of $Gl(K)$ on on the state variables and related quantities, so that only the state space -not the state value- is defined for a given dataset. We show in figure 3(b) the representation produced by equation (12).

What Are the Task-related Patterns Present in the State-Space? Typical signals of activation may be obtained through projection of the state vectors on a suitable basis of vectors; although the choice of a basis derived from typical hemodynamic is possible, we can simply use a basis that comprises delayed stimuli time courses. Doing so, we have obtained two main components which are represented on figure 3 (c). By inspecting the individual response of each voxel to either of the time courses given in 3 (c), it is possible to derive activation maps from the dataset: in figure 3 (bottom), we present two activation maps: one for the voxels that respond to either of the stimuli, one for the voxels that respond to the second stimulus (moving texture) *uniquely*.

Are There Motion-related Artifacts in the Data? Another question of interest is the effect of body motion on the data. To study this, we can simply correlate the state and motion spaces, i.e. the six rigid realignment parameters estimated by the software of Roche et al. (see [3] for related work; SPM motion estimates correlate systematically to brain activation, and thus are not used here). The projection of the motion estimation onto the state space shows that two components can be viewed as common; they are displayed in figure 3(d). Fortunately, they are not correlated with task-related activity.

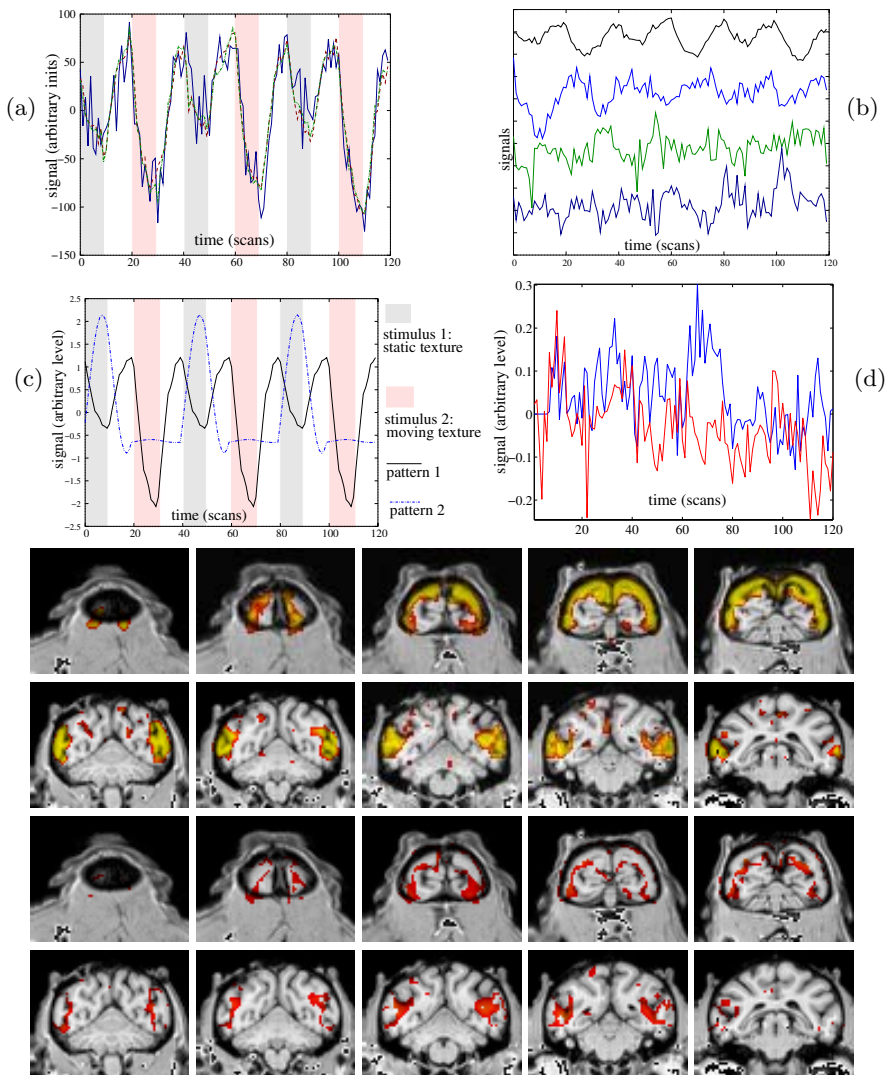


Fig. 3. (a) Approximation of a time course of interest (blue) by the state model: (green, dashed) Reconstructed time course, after reduction to $N = 20$ components by PCA; (red, dashed-dotted) the same, after reduction to $N = 40$ components by PCA. (b) A basis of signals that generate the state-space. While the first one clearly represents task-related activity, there is no such evidence for the other signals. (c) Two main task-related patterns obtained by projection of the state space onto a set of delayed stimulus time courses. They represent the activation patterns elicited in linear, time invariant way, by the stimulation. (d) Two patterns are common to both the state space and the motion estimates. Bottom: Activation maps associated to either of the experimental conditions (first two rows), and to the second condition uniquely (last two rows.) Both maps are thresholded at the significance level $P=10^{-3}$. There is no room here for studying the maps, but they are coherent with standard analyses.

5 Discussion

Links with Existing Methods: Autocorrelation Maximization, Temporal ICA. The state space model can be related to multivariate techniques as ICA. The main difference concerns the fact that usual ICA techniques concentrate on the unmixing problem, not on the evolution problem. The incorporation of temporal information (the experimental paradigm) in the state space model allows for a better detection of task-related signal. This is true especially in the case of event-related experimental designs, where detection is challenging. But the state space model is also flexible, since it does not impose a model for the hemodynamic response. It yields a close approximation of the latter (other methods are more convenient for a precise estimation of the hemodynamic response).

The solution of the estimation problem is in fact very close to the CCA approach proposed in [4], which searches the most strongly autocorrelated patterns of the dataset. The differences are that *i)* our approach can take into account the experimental paradigm *ii)* we test for the rank of the resulting representation *iii)* we are not limited to first order processes, *iv)* we are not limited by the condition $N \ll T$.

Our approach addresses a question that is recurrent in the fMRI multivariate analysis literature: how many components are necessary to adequately describe the data? Existing solutions include the Bartlett test, that amounts to assuming that the signals of non interest have a spherical covariance structure; we believe that this not necessarily sufficient to characterize signals of interest. Our solution is to test whether the temporal structure of the dataset is more autocorrelated than the structure of random datasets. Our experiment on synthetic data shows that this is a valid way of proceeding.

Some Challenges for the Linear Method. The algorithm that we propose for the solution of the problem is quick and efficient; however, one could expect that the EM Kalman method proposed in [6] could perform better. We could never obtain any significant difference between our linear estimation procedure and the outcome of the EM Kalman method on synthetic or real data.

Another still unsolved question is the optimal state representation. It amounts to finding the most meaningful interpretation of the state. Possible solutions involve temporal ICA on the state itself, spatial ICA on the columns of the mixing matrix.

Extensions and Future Work. We have considered a linear model for both the mixing and evolution equations. More general approaches are possible, that use explicitly nonlinear modeling [12]. However, a characteristic of fMRI data is its relatively low signal to noise ratio, so that the introduction of nonlinearities may yield overfitting, and thus misleading interpretation of stochastic components. A nonlinear mixing model is perhaps more affordable; among others, Soatto and Chiuso suggest the use of kernel PCA. An overcomplete representation [7] is also potentially interesting .

Last, a possible extension consists in dealing with multimodal spatio-temporal data: fMRI/MEEG for example. We are aware of the difficulty of mixing the data from different modalities, but we suggest a state-model with two layers: first, a state space describing neural processes, and directly related to the MEEG temporal information that produces a second one that describes the fMRI temporal processes, related to the fMRI data as in this article. But this sketch overlooks many difficulties of the question.

Acknowledgment. Work partially supported by INRIA ARC fMRI and European Grant Mapawamo, No QLG3-CT-2000-30161.

We wish to thank Professor G. Orban, D. Fize and W. Vanduffel, who provided us with the functional MR images we used. The work was developed in collaboration with the laboratory of Neurophysiology, K.U.Leuven, Medical School, Leuven, Belgium (LEUNEURO), directed by G. Orban.

References

1. A. H. Andersen, D.M. Gash, and M.J. Avison. Principal Components Analysis of the Dynamic Response Measured by fMRI: a Generalized Linear Systems Framework. *Magnetic Resonance Imaging*, 17(6):795–815, 1999.
2. V.D. Calhoun, T. Adali, G.D. Pearlson, and J.J. Pekar. Spatial and Temporal Independent Component Analysis of Functional MRI Data Containing a Pair of Task-Related Waveforms. *Human Brain Mapping*, 13:43–53, 2001.
3. L. Freire, A. Roche, and J.F. Mangin. What is the Best Similarity Measure for Motion Correction in fMRI Time Series? *IEEE Transactions on Medical Imaging*, 21(5):470–484, May 2002.
4. Ola Friman, Magnus Borga, Peter Lundberg, and Hans Knutsson. Exploratory fMRI Analysis by Autocorrelation Maximization. *NeuroImage*, 16:454–464, 2002.
5. K.J. Friston, A.P. Holmes, J.B. Poline, P.J. Grasby, S. Williams, R.S. Frackowiak, and R. Turner. Analysis of fMRI time series revisited. *NeuroImage*, 2:45–53, 1995.
6. Zoubin Ghahramani and Geoffrey Hinton. Parameter Estimation for Linear Dynamical Systems. Technical Report CRG-TR-96-2, University of Toronto, 1996.
7. Michael Lewicki and Terrence J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12:337–365, 2000.
8. Martin J. McKeown, S. Makeig, et al. Analysis of fMRI data by blind separation into independent spatial components. *Human Brain Mapping*, 6:160–188, 1998.
9. Sam Roweis and Zoubin Ghahramani. A unifying review of linear gaussian models. *Neural Computation*, 11:305–345, 1999.
10. Stefano Soatto and Alessandro Chiuso. Dynamic data factorization. Technical Report 010001, Department of Computer Science, UCLA, March 2001.
11. Bertrand Thirion and Olivier Faugeras. Dynamical components analysis of fMRI data. In *Proceedings of the 2002 IEEE International Symposium on Biomedical Imaging*, pages 915–918, July 2002.
12. Harri valpola and Juha Karhunen. An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural Computation*, 14:2647–2692, 2002.
13. Wim Vanduffel, Denis Fize, Joseph B. Mandeville, Koen Nelissen, Paul Van Hecke, Bruce R. Rosen, Roger B.H. Tootell, and G. Orban. Visual motion processing investigated using contrast-enhanced fmri in awake behaving monkeys. *Neuron*, 2001. in press.

Tensor Field Regularization Using Normalized Convolution

Carl-Fredrik Westin¹ and Hans Knutsson²

¹ Laboratory of Mathematics in Imaging,
Brigham and Women's Hospital,
Harvard Medical School,
Boston MA, USA
`westin@bwh.harvard.edu`

² Department of Biomedical Engineering,
Linköping University Hospital,
Linköping, Sweden
`knutte@imt.liu.se`

Abstract. This paper presents a filtering technique for regularizing tensor fields. We use a nonlinear filtering technique termed normalized convolution [Knutsson and Westin 1993], a general method for filtering missing and uncertain data. In the present work we extend the signal certainty function to depend on locally derived certainty information in addition to the *a priori* voxel certainty. This results in reduced blurring between regions of different signal characteristics, and increased robustness to outliers. A driving application for this work has been filtering of data from Diffusion Tensor MRI.

1 Introduction

This paper presents a filtering technique for regularizing vector and higher order tensor fields. In particular we focus on filtering of volume data from Diffusion Tensor Magnetic Resonance Imaging (DT-MRI). Related works include [20,19,16,22,21,15,17,6,4,1].

DT-MRI is a relatively recent imaging modality that calls for multi-valued methods for data restoration. DT-MRI measures the diffusion of water in biological tissue. Diffusion is the process by which matter is transported from one part of a system to another owing to random molecular motions. The transfer of heat by conduction is also due to random molecular motion. The analogous nature of the two processes was first recognized by [7], who described diffusion quantitatively by adopting the mathematical equation of heat conduction derived some years earlier by [8]. Anisotropic media such as crystals, textile fibers, and polymer films have different diffusion properties depending on direction. Anisotropic diffusion can be described by an ellipsoid where the radius defines the diffusion in a particular direction. The widely accepted analogy between symmetric 3×3 tensors and ellipsoids makes such tensors natural descriptors

for diffusion. Moreover, the geometric nature of the diffusion tensors can quantitatively characterize the local structure in tissues such as bone, muscle, and white matter of the brain. Within white matter, the mobility of the water is restricted by the axons that are oriented along the fiber tracts. This anisotropic diffusion is due to tightly packed multiple myelin membranes encompassing the axon. Although myelination is not essential for diffusion anisotropy of nerves (as shown in studies of non-myelinated garfish olfactory nerves [3]; and in studies where anisotropy exists in brains of neonates before the histological appearance of myelin [23]), myelin is generally assumed to be the major barrier to diffusion in myelinated fiber tracts.

Using conventional MRI, we can easily identify the functional centers of the brain (cortex and nuclei). However, with conventional proton magnetic resonance imaging (MRI) techniques, the white matter of the brain appears to be homogeneous without any suggestion of the complex arrangement of fiber tracts. Hence, the demonstration of anisotropic diffusion in the brain by magnetic resonance has paved the way for non-invasive exploration of the structural anatomy of the white matter *in vivo* [13,5,2,14].

In DT-MRI, the diffusion tensor field is calculated from a set of diffusion-weighted MR images by solving the Stejskal-Tanner equation. There is a physical interpretation of the diffusion tensor which is closely tied to the standard ellipsoid tensor visualization scheme. The eigensystem of the diffusion tensor describes an ellipsoidal isoprobability surface, where the axes of the ellipsoid have lengths given by the square root of the tensor's eigenvalues. A proton which is initially located at the origin of the voxel has equal probability of diffusing to all points on the ellipsoid.

2 Methods

In this section we outline how normalized convolution can be used for regularizing scalar, vector, and higher order tensor fields.

Normalized convolution (NC) was introduced as a general method for filtering missing and uncertain data [10,19]. In NC, a signal certainty, c , is defined for the signal. Missing data is handled by setting this signal certainties to zero. This method can be viewed as locally solving a weighted least squares (WLS) problem, where the weights are defined by signal certainties and a spatially localizing mask.

A local description of a signal, f , can be defined using a weighted sum of basis functions, B . In NC the basis functions are spatially localized by a scalar (positive) mask denoted the “applicability function”, a . Minimizing

$$\left\| W_a W_c (B\theta - f) \right\| \quad (1)$$

results in the following WLS local neighborhood model:

$$f_0 = B(B^* W_a W_c B)^{-1} B^* W_a W_c f, \quad (2)$$

where W_a and W_c are diagonal matrices containing a and c respectively, and B^* is the conjugate transpose of B .

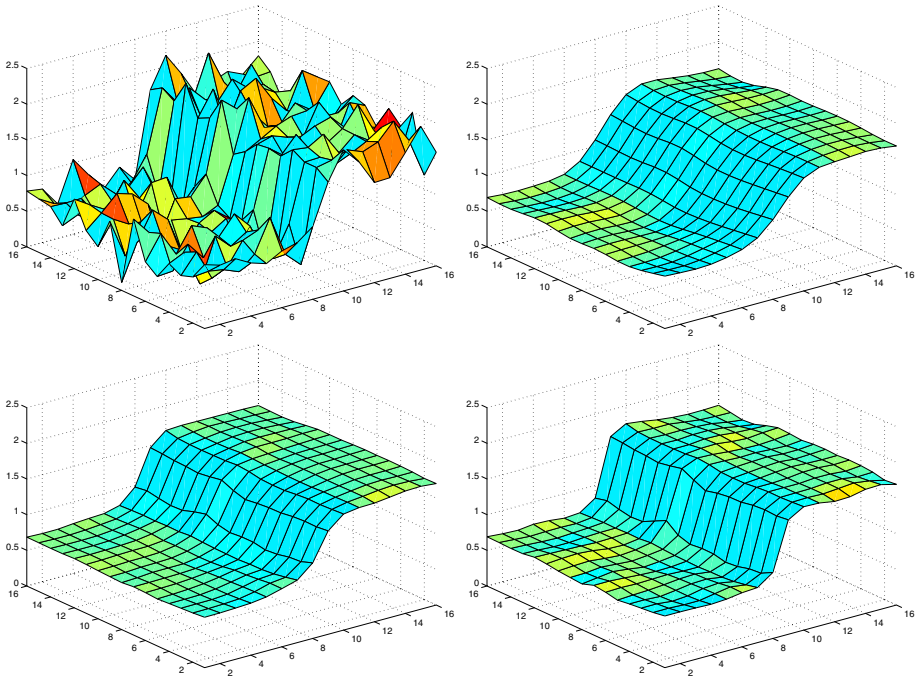


Fig. 1. Filtering of a scalar signal: Original scalar field (upper left) and the result without using the magnitude difference certainty, c_m (upper right). The amount of inter region averaging can be controlled effectively by including this magnitude certainty measure. The smaller the sigma, the smaller inter the region averaging: lower left $\sigma = 1$, lower right $\sigma = 0.5$.

2.1 Certainty Measures

In the present work, a regularization application, the local certainty function, c , consists of two parts:

1. A voxel certainty measure, c_v , defined by the input data.
2. A model/signal similarity measure, c_s :

$$c_s = g(T_0, T),$$

where T_0 is the local neighborhood model. For simplicity we have constructed c_s as a product of separate magnitude and angular similarity measures, c_m and c_a :

$$c_s = c_m c_a.$$

For the magnitude certainty a Gaussian magnitude function has been used in our examples below:

$$c_m = \exp \left[- \left(\frac{|T_0| - |T|}{\sigma} \right)^2 \right]. \quad (3)$$

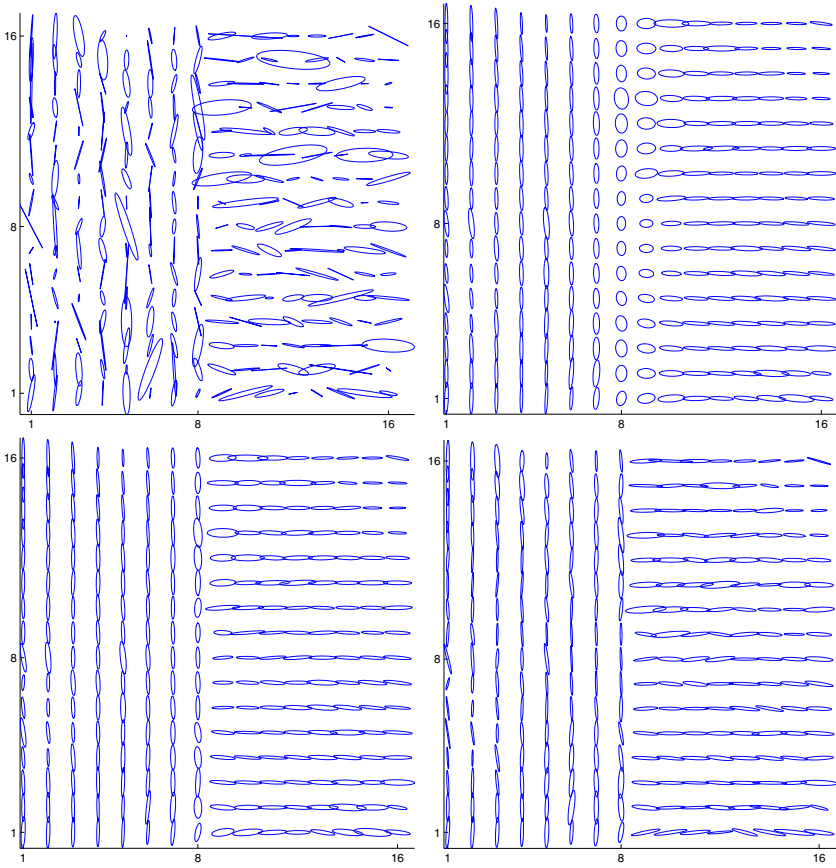


Fig. 2. Tensor field filtering: Original tensor field (upper left) and the result using the proposed method using $\alpha = 0$ (upper right), $\alpha = 2$ (lower left), and $\alpha = 8$ (lower right). Notice how the amount of mixing of tensors of different orientation can be controlled by the angular similarity measure.

The angular similarity measure, c_a , is based on the inner product between the normalized tensors:

$$c_a = \langle \hat{T}_0, \hat{T} \rangle^\alpha,$$

where $\hat{T} = T/|T|$.

The final certainty function is calculated as the product of the voxel certainty and the similarity certainty:

$$c = c_v c_s \quad (4)$$

In general the voxel certainty function, c_v , will be based on prior information about the data. The voxel certainty is set to zero outside the signal extent

to reduce unwanted border effects. If no specific local information is available the voxel certainty is set to one. As described above, the second certainty component, c_s , is defined locally based on neighboring information. The idea here is to reduce the impact of outliers, where an outlier is defined in terms of the local signal neighborhood, and to reduce the blurring across interfaces between regions having very different signal characteristics.

2.2 Simple Local Neighborhood Model

The simplest possible model in the normalized convolution framework is to use only one constant basis function, simplifying the expression for normalized convolution to [10]:

$$T_0 = \frac{\langle a_0, c_v T \rangle}{\langle a_0, c_v \rangle} \quad (5)$$

To focus on the power of introducing the signal/model similarity certainty measure, this simple local neighborhood model is used in our examples below. The applicability function a_0 was set a Gaussian function with standard deviation of 0.75 sample distances.

3 Scalar Field Regularization

In this section we present a scalar example to show the effect of the the voxel and magnitude certainty functions. This concept can be seen as generalization of bilateral filtering [16] into the signal-certainty framework of normalized convolution.

Figure 1 shows the result of filtering a scalar signal using the proposed technique. The upper left plot shows the original scalar signal: a noisy step function. The upper right plot shows the result using standard normalized convolution demonstrating that reduction of noise is achieved at the expense of unwanted mixing of features from adjacent regions. The amount of border blurring can be controlled effectively by including the new magnitude certainty measure, c_m (equation 3). The smaller the sigma, the smaller the inter region averaging as shown by the lower left ($\sigma = 1$) and lower right ($\sigma = 0.5$) plots.

4 Tensor Field Regularization

4.1 Synthetically Generated Tensor Field

Figure 2 shows the result of filtering a synthetic 2D tensor field visualized using ellipses. The original tensor field is shown in the upper left plot. In this example, the voxel certainty measure, c_v , was set to one except outside the signal extent where it was set to zero. The applicability function a was set to a Gaussian function with standard deviation of 3 sample distances.

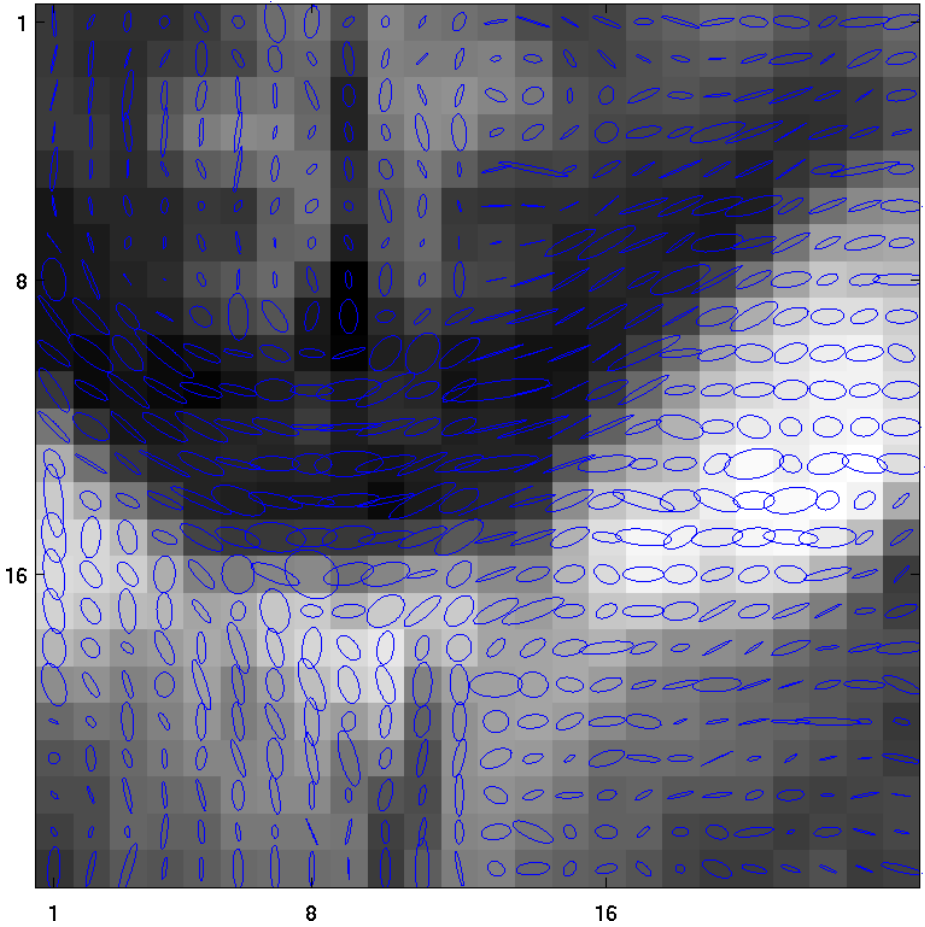


Fig. 3. Original tensor field generated from DT-MRI data.

When filtering tensor data, the angular measure c_a is important since it can be used to reduce mixing of information from regions having different orientations. This is demonstrated in figure 2 using $\alpha = 0$ (upper right), $\alpha = 2$ (lower left), and $\alpha = 8$ (lower right). Notice how the degree of mixing depends on the angular similarity measure.

4.2 Diffusion Tensor MRI Data

Figure 3 shows a tensor field generated from DT-MRI data. In this work we applied a version of the Line Scan Diffusion Imaging (LSDI) technique [9,11, 12]. This method, like the commonly used diffusion-sensitized, ultrafast, echo-planar imaging (EPI) technique [18] is relatively insensitive to bulk motion and physiologic pulsations of vascular origin.

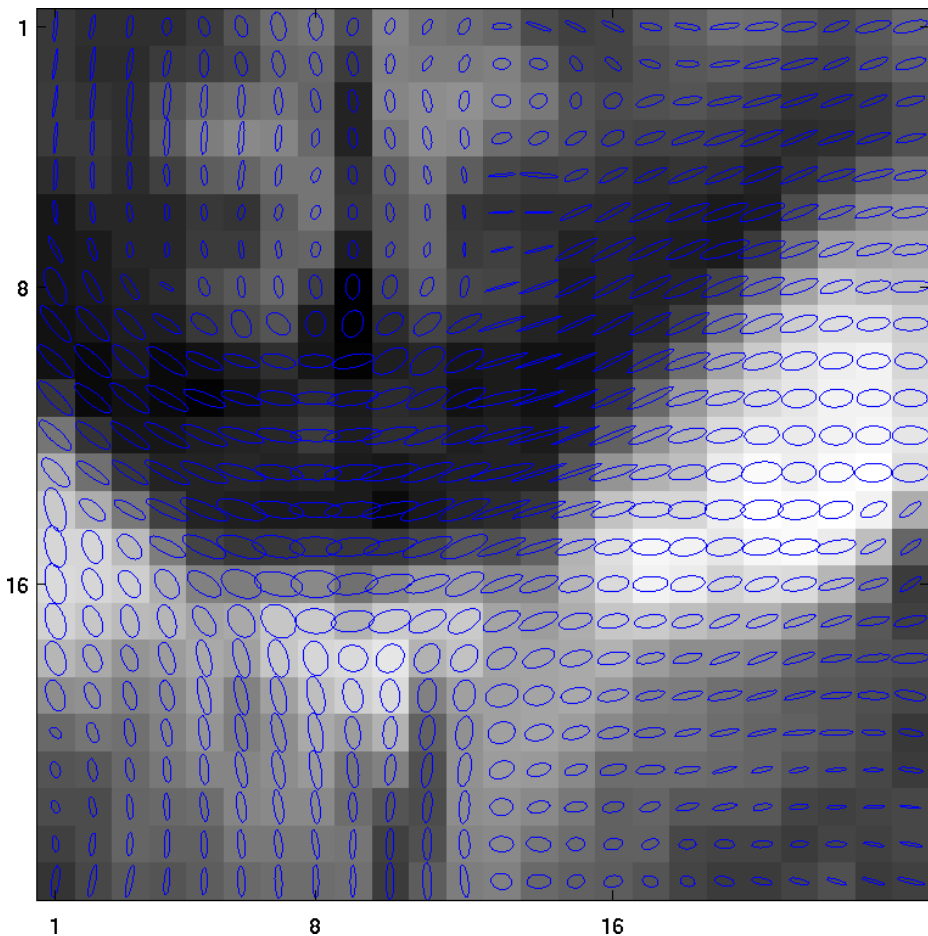


Fig. 4. Result of filtering the DT-MRI tensor field using the proposed method.

The DT-MRI data were acquired at the Brigham and Women's Hospital on a GE Signa 1.5 Tesla Horizon Echospeed 5.6 system with standard 2.2 Gauss/cm field gradients. The time required for acquisition of the diffusion tensor data for one slice was 1 min; no averaging was performed. Imaging parameters were: effective TR=2.4 s, TE=65 ms, $b_{\text{high}}=1000 \text{ s/mm}^2$, $b_{\text{low}}=5 \text{ s/mm}^2$, field of view 22 cm, effective voxel size $4.0 \times 1.7 \times 1.7 \text{ mm}^3$, 4 kHz readout bandwidth, acquisition matrix 128×128 .

Figure 4 shows the result of filtering the DT-MRI tensor field in figure 3 using the proposed method. In this example, the voxel certainty measure, c_v , was set to one except outside the signal extent where it was set to zero. An alternative to this is to use for example Proton Density MRI data defining where the MR signal is reliable. For the angular certainty function, c_a , $\alpha = 4$ was used. The applicability function a was set to a Gaussian function with standard deviation of 3 sample distances.

Acknowledgments. This work was funded in part by NIH grant P41-RR13218, R01-MH 50747 and CIMIT.

References

1. D. Barash. A fundamental relationship between bilateral filtering, adaptive smoothing and the nonlinear diffusion equation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(6):884–847, 2002.
2. P.J. Basser. Inferring microstructural features and the physiological state of tissues from diffusion-weighted images. *NMR in Biomedicine*, 8:333–344, 1995.
3. C. Beaulieu and P.S. Allen. Determinants of anisotropic water diffusion in nerves. *Magn. Reson. Med.*, 31:394–400, 1994.
4. C. Ched'hotel and O. Faugeras D. Tschumperl'e, R. Deric. Constrained flows of matrix-valued functions: Application to diffusion tensor regularization. In *ECCV*, pages 251–265, Copenhagen, June 2002.
5. T.L. Chenevert, J.A Brunberg, and J.G. Pipe. Anisotropic diffusion in human white matter: Demonstration with MR techniques in vivo. *Radiology*, 177:401–405, 1990.
6. O. Coulon, D.C. Alexander, and S.R. Arridge. A regularization scheme for diffusion tensor magnetic resonance images. In *XVIth IPMI, Lecture Notes in Computer Science*, volume 2082, pages 92–105. Springer-Verlag, 2001.
7. A. Fick. Über diffusion. *Ann Phys*, pages 94–59, 1855.
8. J. Fourier. *Théorie analytique de la chaleur*. Académie des Sciences, 1822.
9. H. Gudbjartsson, S.E. Maier, R.V. Mulkern, I. Á. Mórocz, S. Patz, and F.A. Jolesz. Line scan diffusion imaging. *Magn. Reson. Med.*, 36:509–519, 1996.
10. H. Knutsson and C-F. Westin. Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data. In *Proceedings of Computer Vision and Pattern Recognition*, pages 515–523, New York City, USA, June 1993. IEEE.
11. S.E. Maier and H. Gudbjartsson. Line scan diffusion imaging, 1998. USA patent #5,786692.
12. S.E. Maier, H. Gudbjartsson, S. Patz, L. Hsu, K.-O. Lovblad, R.R. Edelman, S. Warach, and F.A. Jolesz. Line scan diffusion imaging: Characteriazation in healthy patients and strok patients. *Am. J. Roentgen*, 17(1):85–93, 1998.
13. M. E. Moseley, Y. Cohen, J. Kucharczyk, J. Mintorovitch, H. S. Asgari, M. F. Wendland, J. Tsuruda, and D. Norman. Diffusion-weighted MR imaging of anisotropic water diffusion in the central nervous system. *Radiology*, 176:439–445, 1990.
14. C. Pierpaoli, P. Jezzard, P. J. Basser, A. Barnett, and G. Di Chiro. Diffusion tensor MR imaging of the human brain. *Radiology*, 201:637, 1996.
15. C. Poupon, C. A. Clark, F. Frouin, J. Régis, I. Bloch, D. Le Bihan, I. Bloch, and J.-F. Mangin. Regularization of diffusion-based direction maps for the tracking brain white matter fascicles. *NeuroImage*, 12:184–195, 2000.
16. C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proc. of ICCV 98*, Bombay, India, 1998.
17. D. Tschumperl'e and R. Deric. Diffusion tensor regularization with constraints preservation. In *CVPR*, Hawaii, 2001.
18. R. Turner, D. le Bihan, J. Maier, R. Vavrek, L. K. Hedges, and J. Pekar. Echo planar imaging of intravoxel incoherent motions. *Radiology*, 177:407–414, 1990.

19. C.-F. Westin. *A Tensor Framework for Multidimensional Signal Processing*. PhD thesis, Linköping University, Sweden, S-581 83 Linköping, Sweden, 1994. Dissertation No 348, ISBN 91-7871-421-4.
20. C.-F. Westin and H. Knutsson. Extraction of local symmetries using tensor field filtering. In *Proceedings of 2nd Singapore International Conference on Image Processing*. IEEE Singapore Section, September 1992.
21. C.-F. Westin, S. E. Maier, H. Mamata, A. Nabavi, F. A. Jolesz, and R. Kikinis. Processing and visualization of diffusion tensor mri. *Medical Image Analysis*, 6(2):93–108, 2002.
22. C.-F. Westin, S.E. Maier, B. Khidhir, P. Everett, F.A. Jolesz, and R. Kikinis. Image Processing for Diffusion Tensor Magnetic Resonance Imaging. In *Medical Image Computing and Computer-Assisted Intervention*, Lecture Notes in Computer Science, pages 441–452, September 1999.
23. D. M. Wimberger, T. P. Roberts, A. J. Barkovich, L. M. Prayer, M. E. Moseley, and J. Kucharczyk. Identification of “premyelination” by diffusion-weighted MRI. *J. Comp. Assist. Tomogr.*, 19(1):28–33, 1995.

Volumetric Texture Description and Discriminant Feature Selection for MRI

Abhir Bhalerao and Constantino Carlos Reyes-Aldasoro

Department of Computer Science,
Warwick University, Coventry, UK
{abhir,creyes}@dcs.warwick.ac.uk

Abstract. This paper considers the problem of texture description and feature selection for the classification of tissues in 3D Magnetic Resonance data. Joint statistical measures like grey-level co-occurrence matrices (GLCM) are commonly used for analysis texture in medical imaging because they are simple to implement but are prohibitively expensive to compute when extended to 3D. Furthermore, the issue of feature selection which recognises the fact that some features will be either redundant or irrelevant is seldom addressed by workers in texture classification. In this work, we develop a texture classification strategy by a sub-band filtering technique similar to a Gabor decomposition that is readily and cheaply extended to 3D. We further propose a generalised sequential feature selection method based on a measure of feature relevance that reduces the number of features required for classification by selecting a set of *discriminant* features conditioned on a set training texture samples. We describe and illustrate the methodology by quantitatively analysing a variety of images: synthetic phantom data, natural textures, and MRI of human knees.

Keywords: Texture classification, Sub-band filtering, Feature selection, GLCM.

1 Introduction

The labelling of tissues in medical imagery such as Magnetic Resonance Imaging (MRI) has rightly received a great deal of attention over the past decade. Much of this work has concentrated on the classification of tissues by grey level contrast alone. For example, the problem of grey-matter white-matter labelling in central nervous system (CNS) images like MRI head-neck studies of has been achieved with some success by supervised statistical classification methods, notably EM-MRF [25]. Some of this success is partly as a result of incorporating MR bias-field correction into the classification process. One can regard this as extending the image model from a simple piece-wise-constant with noise to include a slowly varying additive or multiplicative bias to the image grey-levels [23]. Another reason why first-order statistics have been adequate in many instances is that the MR imaging sequence can be adapted or tuned to increase contrast in the tissues

of interest. For example, a T2 weighted sequence is ideal for highlighting cartilage in MR orthopaedic images, or the use of iodinated contrast agents for tumours and vasculature. Multimodal image registration enables a number of separately acquired images to be effectively fused to create a multichannel or multispectral image as input to a classifier. Other than bias field artefact, the ‘noise’ in the image model incorporates variation of the voxel grey-levels due to the *textural* qualities of the imaged tissues and, with the ever increasing resolution of MR scanners, it is expedient to model and use this variation, rather than subsuming it into the image noise.

The machine vision community has extensively researched the description and classification of 2D textures, but even if the concept of image texture is intuitively obvious to us, it can be difficult to provide a satisfactory definition. Texture relates to the surface or structure of an object and depends on the relation of contiguous elements and may be characterised by granularity or roughness, principal orientation and periodicity (normally associated with man-made textures such as woven cloth). The early work of Haralick [8] is the standard reference for statistical and structural approaches for texture description. Other approaches include contextual methods like Markov Random Fields as used by Cross and Jain [4], and fractal geometry methods by Keller [10]. Texture features derived from the grey level co-occurrence matrix (GLCM) calculate the joint statistics of grey-levels of pairs of pixels at varying distances (limited by the matrix size) and is a simple and widely used texture feature. Unfortunately, the matrix size, M^d for a d dimensional image size N makes the descriptor have complexity of $O(N^d M^d)$ prohibitive for $d = 3$. For these reasons and to capture the spatial-frequency variation of textures, filtering methods akin to Gabor decomposition [24] and joint spatial/spatial-frequency representations like Wavelet transforms have been reported (e.g. [12]). Randen [17] has shown that co-occurrence measures are outperformed by such filtering techniques. The dependence of texture on resolution or scale has been recognised and exploited by workers which has led to the use of multiresolution representations such as the Gabor decomposition and the wavelet transform [12] [13]. Here we use the Wilson-Spann sub-band filtering approach [26] which is similar to the Gabor filtering and has been proposed as a ‘complex’ wavelet transform [19].

The importance of texture in MRI has been the focus of some researchers, notably Lerksi [6] and Schad [22], and a COST European group has been established for this purpose [3]. Texture analysis has been used with mixed success in MRI, such as for detection of microcalcification in breast imaging [5] and for knee segmentation [9], and in CNS imaging to detect macroscopic lesions and microscopic abnormalities such as for quantifying contralateral differences in epilepsy subjects [20], to aid the automatic delineation of cerebellar volumes [15] and to characterise spinal cord pathology in Multiple Sclerosis [14]. Most of this reported work, however, has employed solely 2D measures, again based on GLCM. Furthermore, feature selection is often performed in an empirical way with little regard to training data which are usually available.

In this paper we describe a fully 3D texture description scheme using a multiresolution sub-band filtering [26] and to develop a strategy for selecting the most *discriminant* texture features conditioned on a set of training images containing examples of the tissue types of interest. The ultimate goal is to select a compact and appropriate set of features thus reducing the computationally burden in both feature extraction and subsequent classification. We describe the 2D and 3D frequency domain texture feature representation and the feature selection method, by illustrating and quantitatively comparing results on example 2D images and 3D MRI.

2 Multiresolution Sub-band Filtering

Textures can vary in their spectral distribution in the frequency domain, and therefore a set of sub-band filters can help in their discrimination: if the image contains textures that vary in orientation and frequency, then certain filter sub-bands will be more energetic than others, and ‘roughness’ will be characterised by more or less energy in broadly circular band-pass regions. Wilson and Spann [26] proposed a set of operations that subdivide the frequency domain of an image into smaller regions by the use of compact and optimal (in spatial versus spatial-frequency energy) filter functions based on finite prolate spheroidal sequences (FPSS). For ease of implementation, we approximated these functions with truncated Gaussians (figure 1) essentially creating a band-limited Gabor filter basis where each filter or feature estimates energy in particular frequency bands.

A variety of tessellations of these sub-band filters are possible. For this work, the Second Orientation Pyramid (SOP) arrangement presented in figure 2 was selected for the tessellation of the frequency domain. The SOP tessellation involves a set of 7 filters, one for the low-pass region and six for the high-pass. The centred Fourier transform $\mathcal{I}_\omega = \mathcal{F}\{\mathcal{I}\}$ of a given image $\mathcal{I}$ can be subdivided into a set of non-overlapping regions $L_r^i \times L_c^i$. $L_r^i = \{r, r+1, \dots, r+N_r^i\}$, $1 \leq r \leq N_r - N_r^i$, $L_c^i = \{c, c+1, \dots, c+N_c^i\}$, $1 \leq c \leq N_c - N_c^i$. Using these coordinate sets, we can write the i th subdivision or band of the frequency domain using the SOP tessellation as:

$$L_r \times L_c; F^i : \begin{cases} L_r^i \times L_c^i & \rightarrow N(\mu^i, \Sigma^i) \\ (L_r^i \times L_c^i)^c & \rightarrow 0 \end{cases} \quad \forall i \in SOP \quad (1)$$

where μ^i is the centre of the region i and Σ^i is the variance of the Gaussian that will provide a cut-off of 0.5 at the limit of the band (figure 1). Then, the i th feature S_ω^i in its frequency and spatial domains is calculated by the convolution:

$$S_\omega^i(k, l) = F^i(k, l)I_\omega(k, l) \quad \forall (k, l) \in (L_r \times L_c), \quad S^i = \mathcal{F}^{-1}\{S_\omega^i\} \quad (2)$$

For a pyramid of order 2, the central region $(L_r^1(1) \times L_c^1(1))$ is itself sub-divided into central region of half the size $(L_r(2) \times L_c(2))$ with dimensions $N_r(2) = \frac{N_r(1)}{2}$, $N_c(2) = \frac{N_c(1)}{2}$ with surrounding regions $\frac{1}{4}$ of the size (figure 2). It is

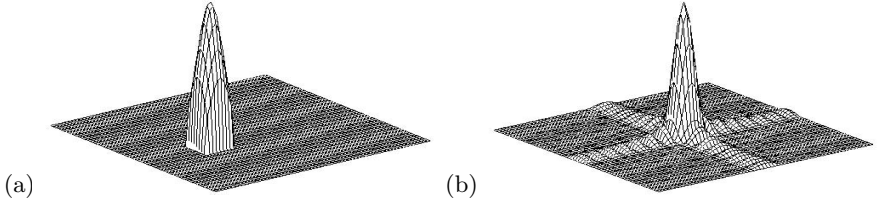


Fig. 1. Band-limited Gaussian Filter $F^i = N(\mu_i, \Sigma_i)$: (a) Frequency domain, (b) Spatial Domain.

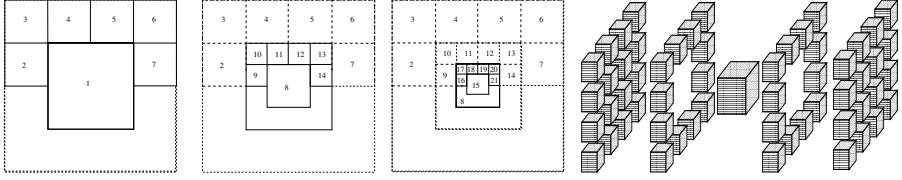


Fig. 2. 2D and 3D Second Orientation Pyramid (SOP) tessellation. Solid lines indicate the filters added at the present order while dotted lines indicate filters added in lower orders. (a) 2D order 1, (b) 2D order 2, (c) 2D order 3, and (d) 3D order 1.

assumed that $N_r(1) = 2^a$, $N_c(1) = 2^b$ so that the results of the divisions are always integer values.

From a computational point of view, the process of subdivision can be easily performed by the combination of two operators, the *quadrant operator* and the *centre-surround operator*. The quadrant operator divides the frequency domain into four quadrants, and the centre-surround operator, which splits into an inner square region and a surrounding annulus. The dimension for the quadrants are $Q1\{1 \dots \frac{N_r}{2}, 1 \dots \frac{N_c}{2}\}$, $Q2\{1 \dots \frac{N_r}{2}, \frac{N_c}{2} + 1 \dots \frac{N_c}{2}\}$, $Q3\{\frac{N_r}{2} + 1 \dots \frac{N_r}{2}, 1 \dots \frac{N_c}{2}\}$, $Q4\{\frac{N_r}{2} + 1 \dots \frac{N_r}{2}, \frac{N_c}{2} + 1 \dots \frac{N_c}{2}\}$, and for the centre-surround $C\{\frac{N_r}{2} + 1 - \frac{N_r}{x} \dots \frac{N_r}{2} + \frac{N_r}{x}, \frac{N_c}{2} + 1 - \frac{N_c}{x} \dots \frac{N_c}{2} + \frac{N_c}{x}\}$, $S\{C^c\}$. Where x determines the dimensions of the inner centre, if $x = 4$, the width of the centre is one half of the original image which was used through out this work. Combining the operators in diverse combinations and order can yield different tessellations, including the well known Gaussian and Laplacian Pyramids. The decomposition also bears similarities to a complex Wavelet transform [24].

In order to filter a three dimensional set, a 3D tessellation (figure 2(d)) is required. The filters will again be formed by truncated 3D Gaussians in a octave-wise tessellation that resemble a regular oct-tree configuration. In the case of MR data, these filters can be applied directly to the K-space. As in the 2D case, the low pass region will be covered by one filter, but the surround or high pass region is more complicated (again half of the space is not used because of symmetry properties of the DFT). While there are 6 high pass filters in a 2D tessellation, in three dimensions there are 28 filters. This tessellation yields 29 features per order. The definitions of the filters follows the extension of the space of rows and columns to $L_r \times L_c \times L_l$ with the new dimension l - levels.

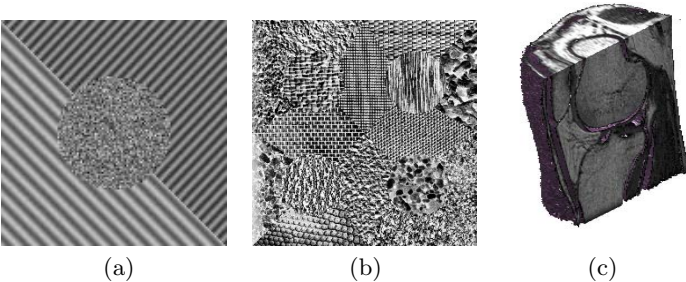


Fig. 3. Example images used in experiments (a) 2D synthetic phantom (2D), (b) Image containing 16 natural textures taken from Brodatz album arrange by Randen and Husøy [17], (c) 3D MRI of human knee containing $512 \times 512 \times 87$ voxels with dimensions $0.25 \times 0.25 \times 1.4\text{mm}$.

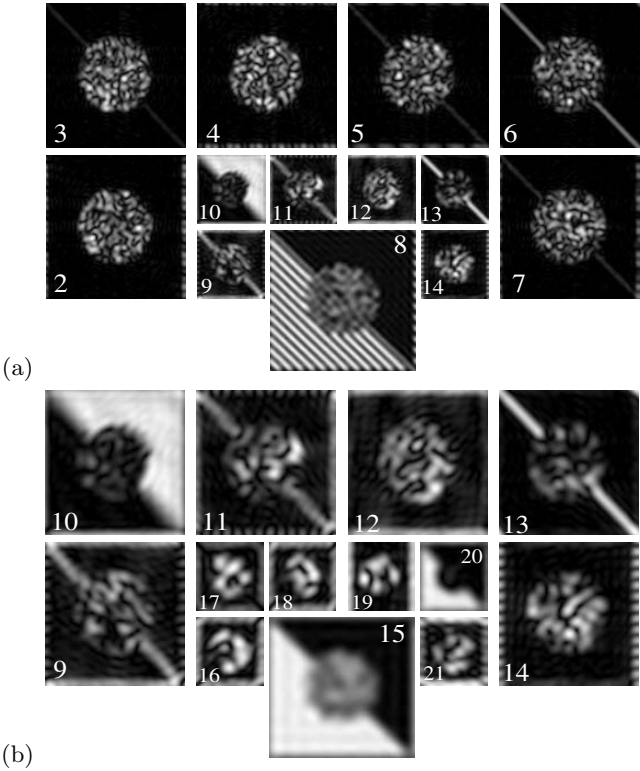


Fig. 4. Two sets of features S^i from the phantom image (a) Features 2 to 14 (Note S^{10} highlights one oriented pattern) (b) Features 9 to 21 (note S^{20} highlights the other oriented pattern). In each set, the feature S^i is placed in the position corresponding to the filter F^i in the frequency domain.

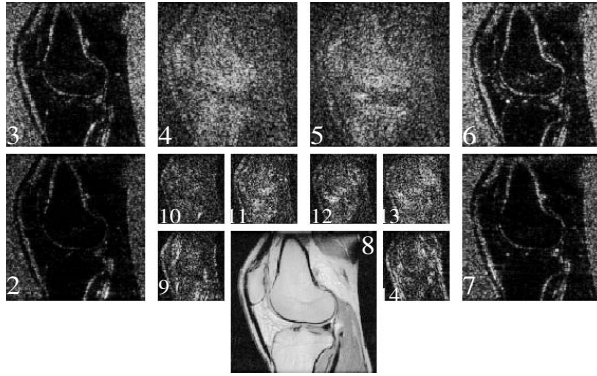


Fig. 5. Features 2 to 14 from one slice of the human knee MRI. The background is well represented in feature images $S^{2,3}$, the bone appears in feature images $S^{4,5,11,12}$ and tissue in $S^{9,13,14}$.

Figure 4 shows the feature space S^i of the 2D synthetic phantom shown in figure 3(a). Figure 4(a) contains the features of orders 1 and 2, and figure 4(b) shows the features of orders 2 and 3. Note how in S^{2-7} , the features that are from high pass bands, only the central region, which is composed of noise, is present. The oriented patterns have been filtered out. S^{10} and S^{20} show the activation due to the oriented patterns. S^8 is a low pass filter and still keeps a trace of one of the oriented patterns.

3 Discriminant Feature Selection

Feature selection is a critical step in classification since not all features derived from sub-band filtering, GLCM, wavelets, wavelet packet or any other methodology have the same discrimination power. When a large number of features are input to a classifier, some may be *irrelevant* while others will be *redundant* - which will at best increase the complexity of the task, and at worst hinder the classification by increasing the inter-class variability. Eigenspace methods like PCA are traditionally used for feature selection where the feature space is transformed to a set of independent and orthogonal axes which can be ranked by the extent of variation given by the associated eigenvalues. Fisher's linear discriminant analysis (LDA) on the other hand, finds the feature space mapping which maximises the ratio of between-class to within-class variation jointly for each feature (dimension) [7] given a set of training data. PCA can be further applied to find a *compact* subspace to reduce the feature dimensionality. However, while these eigenspace methods are optimal and effective, they still require the computation of all the features for given data.

We propose a supervised feature selection methodology based on the discrimination power or *relevance* of the individual features taken independently, the ultimate goal is select a reduced number m of features or bands (in the 2D case $m \leq 7o$, and in 3D $m \leq 29o$, where o is the order of the SOP tessellation). In

order to obtain a quantitative measure of *how separable* are two classes, a distance measure is required. We have studied a number measures (Bhattacharyya, Euclidean, Kullback-Leibler ([16]), Fisher and have empirically shown that the *Bhattacharyya distance*(BD) works best on a variety of textures [1]. The BD of two classes (a, b) is calculated by

$$BD(a, b) = \frac{1}{4} \ln \left\{ \frac{1}{4} \left(\frac{\sigma_a^2}{\sigma_b^2} + \frac{\sigma_b^2}{\sigma_a^2} + 2 \right) \right\} + \frac{1}{4} \left\{ \frac{(\mu_a - \mu_b)^2}{\sigma_a^2 + \sigma_b^2} \right\} \quad (3)$$

where μ_j, σ_j are the mean and variances of class j . Note that the second term of BD is the Mahalanobis distance (used in Fisher LDA) which works fine as a metric when $\mu_a \neq \mu_b$. However, when the class means are equal, the first term of BD compares the ratios of the class variances.

The feature selector works as follows:

1. The Bhattacharyya distance for sub-band feature i , $BD^i(a, b)$ is calculated for all pairs of $N_p = \binom{n}{2}$ of classes $\mathbf{q} = (a, b)$ picked from n classes where the sample statistics μ_j, σ_j for class j are

$$\mu_j = \sum_{N(\mathcal{I})} S^i(j), \quad \sigma_j^2 = \sum_{N(\mathcal{I})} [S^i(j)]^2 - \mu_j^2 \quad (4)$$

2. The marginal distance for each feature i , across all pairs of training classes N_p

$$M^i = \sum_{\mathbf{q}}^{N_p} BD^i(\mathbf{q}) \quad (5)$$

is then rank ordered such that $M = \{M^1, M^2, \dots, M^{70}\}, M^1 \geq M^2 \geq \dots \geq M^{70}$.

3. Sub-band features are selected from the test data according the marginal rank ordering and fed sequentially into the classifier.

The feature selector is sub-optimal in the sense that there is no guarantee that the selected feature sub-space is the best, but our method does not exclude the use of PCA or LDA to diagonalise the result to aid the classification. Since we discount effect of the classifier from our ranking, the method falls under the *filter* category of Kohavi et al [11]. We note that Boz [2] have proposed a similar method. Saito [21] provides a categorical analysis of such feature selection methods.

The human knee MRI marginal ranking space shown in figure 6(a) was formed with four $32 \times 32 \times 32$ training regions of background, muscle, bone. These training regions, which are small relative to the size of the data set, were manually segmented, and they were included in the classified test data. It can be immediately noticed that two bands ($S^{22,54}$, low-pass) dominate the discrimination while the distance of the pair *bone-tissue* is practically zero compared with the rest of the space. Figure 6 (b) zooms into the Bhattacharyya distances of this

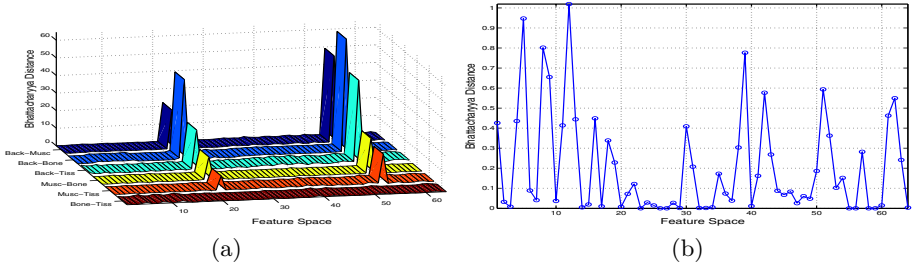


Fig. 6. Human knee MRI (a) Bhattacharyya Space BS (3D, order 2) $\binom{3}{4} = 6$ pairs (b) Bhattacharyya Space($BS^i(\text{bone}, \text{tissue})$).

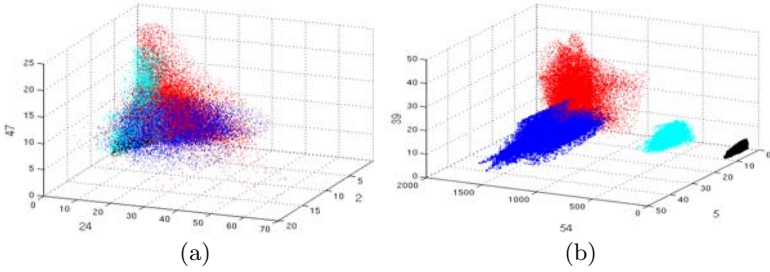


Fig. 7. Scatter plots of three features S^i from human knee MRI (3D, order 2) (a) *bad* discriminating features $S^{2,24,47}$ (b) *good* discriminating features $S^{5,39,54}$. Note that each feature corresponds to a filtered version of the data, therefore the axis values correspond to the magnitude of each feature.

pair. Here we can see that some features: 12, 5, 8, 38, $\dots$, could provide discrimination between bone and tissue, and the low pass bands could help discriminate the remaining classes.

To illustrate the effect of the feature selector on discrimination of clusters in the feature space, figure 7 shows the scatter plot of three features selected at random and three features taken from the beginning of the marginal rank order set, M , based on the Bhattacharyya distance.

4 Experimental Results and Discussion

For every data set the feature space was classified with a K-means classifier, which was selected for simplicity and speed. Features from the multiresolution sub-band filtering using the SOP tessellation were sequentially input to the classifier by the marginal rank order statistic M . The average misclassification error was calculated and plotted against the number of features introduced. For the MRI data, we manually delineated the tissue classes of interest for the error measurement. Note that the dimensionality of the feature space increases by 1 at each step.

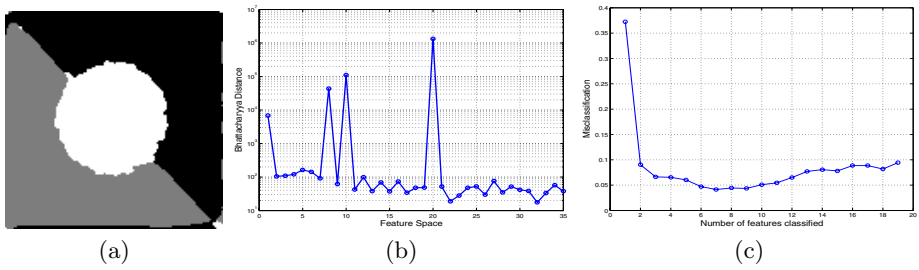


Fig. 8. Classification of the figure 3(a), (a) Classified 2D Phantom at misclassification 4.13%: (b) Marginal Distribution of the Bhattacharyya Space BS^i . (Note the high values for features 10 and 20), (c) Misclassification error against features input to classifier selected by marginal rank order feature selection (b).

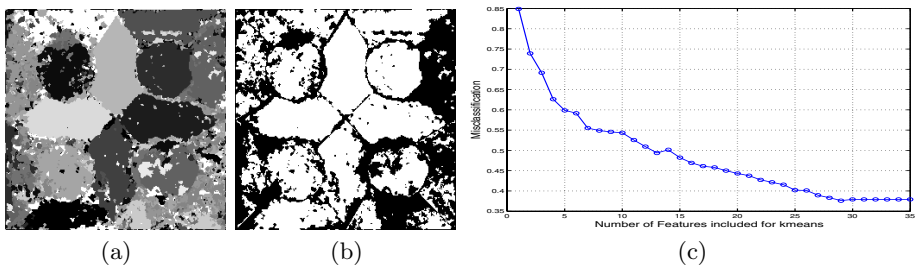


Fig. 9. Classification of the natural textures image (figure 3(b)) with 16 different textures: (a) Classification results at 37.2% misclassification, (b) Pixels correctly classified shown as white, (c) Misclassification error against features input to classifier selected by marginal rank order feature selection (b).

Figures 8 (c) and 9 (c) show the misclassification as the features are included using the sequential discriminant feature selection. Figure 8 (a) shows the classification of the 2D synthetic phantom at 4.3% misclassification with 7 features (out of 35). Of particular note were features 10 and 20 which can be seen in the marginal of the Bhattacharyya space in figure 8 (b). The low-pass features 1 and 8 also have high values but should not be included in this case since they contain the frequency energy that will be disclosed in features 10 and 20 giving more discrimination power. The misclassification plot in figure 8 (c) shows how the first two features manage to classify correctly more than 90% of the pixels and then the next 5, which describe the central circular region, worsen the misclassification. If more features are added, the classification would not improve.

The natural textures image present a more difficult challenge. Randen and Husøy [18] used 9 techniques to classify this image, (they did not report results using the FPSS filtering). For various filtering methods their misclassification results were: Dyadic Gabor filter banks (60.1%), Gabor filters (54.8%), co-occurrence (49.6%), Laws filters (48.3%), Wavelets (38.2%), Quadrature Mirror Filters (36.4%). Our misclassification of SOP filtering is 37.2%, placing this

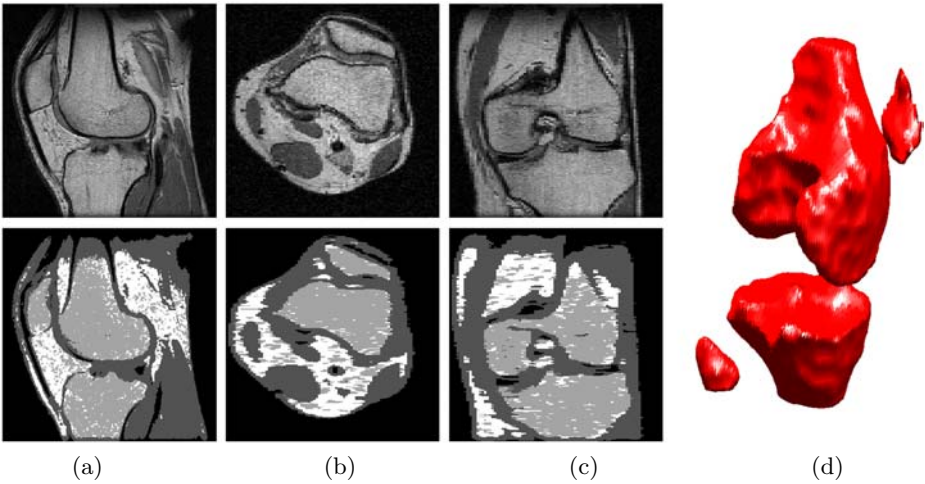


Fig. 10. Supervised classification of human knee MRI into tissue, bone, background and muscle regions using multiresolution sub-band filtering with a SOP tessellation and marginal rank order feature selection taking the 9 most discriminant 3D texture features from a possible of 87 (average misclassification error 8.1%): (a) Sagittal view slice 45, (b) Axial slice 200, (c) Coronal slice 250 and (d) Surface rendering of the bones.

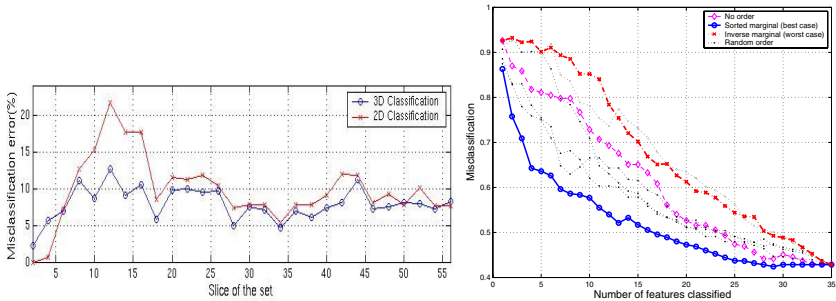


Fig. 11. (a) Comparison of classification errors based on 2D and 3D texture features using sub-band filtering. (b) Empirical analysis of feature selector optimality by comparing the marginal rank order selection against reversed rank order selection and several random orderings. Note how the (decreasing) rank order and reverse rank orders roughly bound the $n!$ possible rank orderings given a choice of n features from which to choose. The convergence of the plots beyond half-way is to be expected since the number of possible orderings reduced dramatically by: $(n/2)! \ll n!$.

in second place. Figure 9(a) shows the final classification and figure 9(b) show the pixels that were correctly classified. Here it becomes clear that several textures are almost completely described by the feature space, and thus are correctly classified, while some textures are not correctly classified, like the rocks at the upper right hand side corner. The misclassification decreases while adding fea-

tures and requires almost all of them in contrast with the synthetic phantom previously described.

The original MRI of the human knee data set consisted of 87 slices of 512×512 pixels each. The classification was performed with the low-pass feature, 54, and the ordered statistics of the bone-tissue feature space: $S^{12,5,8,39,9,51,42,62}$ selected by the marginal rank order selection. This reduced significantly the computational burden since only these 9 features were selected from a possible of 87. The misclassification obtained was 8.1%. Several slices in axial, coronal and sagittal planes with their respective classifications are presented in figure 10. To compare this result with a GLCM based scheme and demonstrate the use of the marginal rank order feature selection, one slice of the human knee MRI set was selected and classified with both methods. The Bhattacharyya discriminant measure was calculated on 10 GLCM features: Contrast $f_2(\theta = 0, \frac{\pi}{2}, \frac{3\pi}{4})$, Inverse difference moment: $f_5(\theta = \frac{3\pi}{4})$, Variance $f_{10}(\theta = 0, \frac{\pi}{2}, \frac{3\pi}{4})$, Entropy $f_{11}(\theta = 0, \frac{\pi}{2}, \frac{3\pi}{4})$, plus the image grey-level. The six most GLCM discriminant features were classified giving an error of 17.0% whereas for the SOP on this slice, a lower error of 7% was achieved. We also compared a 2D SOP run slice-by-slice on the knee MRI against a fully 3D texture descriptor based on the oct-tree SOP tessellation, figure 11(b). This plot shows a noticeable improvement in the missclassification errors using approximately the same number of features in 3D over the slice-by-slice 2D implementation. Whether the extra complexity is ultimately of value remains to be seen since we have not conducted experiments on sufficient MRI data to make a judgement at present.

Finally, it is instructive to try and gauge the optimality of the marginal rank order feature selection. Figure 11(b) shows plots of misclassification error for the Randen natural textures image for different feature order selections: the decreasing rank order selection, the reverse (increasing) rank order selection and several random orderings. What is clear the forward and reverse rank order feature selections appear to bound the distribution of all rank order selections, with the random selections tending to cluster around the middle. If it is to be believed that the majority of the $n!$ possible orderings have a central tendency, then the two bounding orderings are both unlikely to occur by chance. This suggests that the marginal rank ordering is close to the optimal feature selection ordering. We hope to investigate this conjecture further using a texture model.

References

1. A. Bhalerao and N. Rajpoot. Discriminant Feature Selection for Texture Classification. In *Submitted to British Machine Vision Conference BMVC'03*, 2003.
2. O. Boz. Feature Subset Selection by Using Sorted Feature Relevance. In *Proc. Intl. Conf. on Machine Learning and Applications*, June 2002.
3. COST European Cooperation in the field of Scientific and Technical Research. *COST B11 Quantitation of MRI Texture*. <http://www.uib.no/costb11/>, 2002.
4. G. R. Cross and A. K. Jain. Markov Random Field Texture Models. *IEEE Trans. on PAMI*, PAMI-5(1):25–39, 1983.

5. D. James et al. Texture Detection of Simulated Microcalcification Suceptibility Effects in MRI of the Breasts. *J. Mag. Res. Imaging*, 13:876–881, 2002.
6. R.A. Lerski et.al. MR Image Texture Analysis - An Approach to Tissue Characterization. *Mag. Res. Imaging*, 11(6):873–887, 1993.
7. K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, 1972.
8. R. M. Haralick. Statistical and Structural Approaches to Texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.
9. T. Kapur. *Model based three dimensional Medical Image Segmentation*. PhD thesis, AI Lab, Massachusetts Institute of Technology, May 1999.
10. J. M. Keller and S. Chen. Texture Description and Segmentation through Fractal Geometry. *Computer Vision, Graphics and Image Processing*, 45:150–166, 1989.
11. R. Kohavi and G. H. John. Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.
12. S. Livens, P. Scheunders, G. Van de Wouwer, and D. Van Dyck. Wavelets for texture analysis, an overview. In *6th Int. Conf. on Image Processing and its Aplplications*, volume 2, pages 581–585, Dublin, Ireland, july 1997.
13. M. Eden M. Unser. Multiresolution Feature Extraction and Selection for Texture Segmentation. *IEEE Trans. on PAMI*, 11(7):717–728, 1989.
14. J. M. Mathias, P. S. Tofts, and N. A. Losseff. Texture Analysis of Spinal Cord Pathology in Multiple Sclerosis. *Mag. Res. in Medicine*, 42:929–935, 1999.
15. I. J. Namer O. Yu, Y. Mauss and J. Chambron. Existence of contralateral abnormalities revealed by texture analysis in unilateral intractable hippocampal epilepsy. *Magnetic Resonance Imaging*, 19:1305–1310, 2001.
16. N.M. Rajpoot. Texture Classification Using Discriminant Wavelet Packet Subbands. In *Proc. IEEE Midwest Symposium on Circuits and Systems*, Aug. 2002.
17. T. Randen and J. H. Husøy. Filtering for Texture Classification: A Comparative Study. *IEEE Trans. on PAMI*, 21(4):291–310, 1999.
18. T. Randen and J. H. Husøy. Texture segmentation using filters with optimized energy separation. *IEEE Trans. Image Processing*, 8:571–582, 1999.
19. P. De Rivaz and N. G. Kingsbury. Complex Wavelet Features for Fast Texture Image Retrieval. In *Proc. ICIP 1999*, pages 109–113, 1999.
20. N. Saeed and B. K. Piri. Cerebellum Segmentation Employing Texture Properties and Knowledge based Image Processing : Applied to Normal Adult Controls and Patients. *Magnetic Resonance Imaging*, 20:425–429, 2002.
21. N. Saito, R. Coifman, F. B. Geshwind, and F. Warner. Discriminant Feature Extraction Using Empirical Probability Density Estimation and a Local Basis Library. *Pattern Recognition*, 35:2841–2852, 2002.
22. L. R. Schad, S. Bluml, and I. Zuna. MR Tissue Characterization of Intracranial Tumors by means of Texture Analysis. *Mag. Res. Imaging*, 11:889–896, 1993.
23. M. Styner, C. Brechbühler, G. Székely, and G. Gerig. Parametric estimate of intensity inhomogeneities applied to mri. *IEEE Transactions on Medical Imaging*, 19(3):153–165, 2000.
24. M. Unser. Texture Classification and Segmentation Using Wavelet Frames. *IEEE Trans. on Image Processing*, 4(11):1549–1560, 1995.
25. W.M. Wells, W.E.L. Grimson, R. Kikinis, and F.A. Jolesz. Adaptive segmentation of MRI data. *IEEE Trans. on Medical Imaging*, 15(4):429–442, 1996.
26. R. Wilson and M. Spann. Finite Prolate Spheroidal Sequences and Their Applications: Image Feature Description and Segmentation. *IEEE Trans. PAMI*, 10(2):193–203, 1988.

White Matter Mapping in DT-MRI Using Geometric Flows

Lisa Jonasson¹, Patric Hagmann¹, Xavier Bresson¹, Reto Meuli²,
Olivier Cuisenaire¹, and Jean-Philippe Thiran¹

¹ Signal Processing Institute (ITS),
Swiss Federal Institute of Technology (EPFL),
CH-1015 Lausanne, Switzerland
{Lisa.Jonasson,Xavier.Bresson,Patric.Hagmann,
Olivier.Cuisenaire,JP.Thiran}@epfl.ch
<http://ltswww.epfl.ch/~{ }brain>

² Department of Diagnostic and Interventional Radiology,
Lausanne University Hospital (CHUV),
CH-1011 Lausanne, Switzerland
Reto.Meuli@chuv.hospvd.ch

Abstract. We present a 3D geometric flow designed to evolve in Diffusion Tensor Magnetic Resonance Images (DT-MRI) along fiber tracts by measuring the diffusive similarity between voxels. Therefore we define a front propagation speed that is proportional to the similarity between the tensors lying on the surface and its neighbor in the propagation direction. The method is based on the assumption that successive voxels in a tract have similar diffusion properties. The front propagation is implemented using level set methods by Osher and Sethian [1] to simplify the handling of topology changes and provides an elegant tool for smoothing the segmented tracts. While many methods demand a regularized tensor field, our geometrical flow performs a regularization as it evolves along the fibers. This is done by a curvature dependent smoothing term adapted for thin tubular structures. The purpose of our approach is to get a quantitative measure of the diffusion in segmented fiber tracts. This kind of information can also be used for white matter registration and for surgical planning.

1 Introduction

Diffusion tensor MRI (DT-MRI) is a relatively new modality that permits non-invasive quantification of the water diffusion in living tissues. The diffusion tensor provides information about both quantity and directions of the main diffusions at a certain point. The water diffusion in the brain is highly affected by the ordered structures of axons, cell membranes and myelin sheaths. The DT becomes highly anisotropic in these fibrous regions and DT-MRI is therefore a useful tool for locating these kinds of structures, providing important information about the brain connectivity, with potential major impact on fundamental neuroscience as well as on clinical practise for a better understanding of many related diseases

such as multiple sclerosis [2][3], Alzheimer's disease [4] [5], schizophrenia [6][7] and dyslexia [8].

The DT is normally interpreted by calculating its eigenvalues and eigenvectors, the eigenvector corresponding to the highest eigenvalue describes the direction of the principal diffusion and the eigenvalue is a quantitative measure of the diffusion in that direction. Most of the existing methods for tracking fiber bundles rely on this principal diffusion direction to create integral curves describing the fiber paths [9] [10] [11]. Other methods are using a probabilistic approach to explore more of the information contained in the diffusion tensor like for example Hagmann et al. that consider the tensor as a probability distribution [12]. Parker et al. [13] have used level set theory, applied using fast marching methods, to find the connection paths between different brain regions. Campbell et al. [14] have also used level set theory but used it to implement a geometrical flow to track the fiber. They mostly focus on the problem to prevent leakage of the thin tubular structure that represents the fibers, by using flux maximizing flows.

Batchelor et al. [15] are using more of the tensor information by iteratively solving the diffusion equation. The method creates paths that originates from a chosen seed-point, and can be considered as probability measures of a connection. A similar approach is presented by O'Donnell et al. [16] where they find the steady state of the diffusion equation to create a flux vector field. In the same paper they show how the inverse diffusion tensor can define a Riemannian metric that is then used to find geodesic paths that can be interpreted as fiber tracts.

The above methods focus on finding individual fiber paths whereas we have chosen to search for regions corresponding to certain fiber tracts, following the same idea as Tench et al. [17]. For this we use a 3D geometric flow designed to evolve along the fiber tracts by measuring the diffusive similarity between voxels. The front propagation is implemented using level set methods by Osher and Sethian [1]. It simplifies the handling of topology changes and provides an elegant tool for smoothing the segmented tracts. While many methods demand a regularized tensor field, our geometrical flow performs a regularization as it evolves along the fibers. The purpose of our approach is to get a quantitative measure of the diffusion in fiber tracts. This kind of information can also be used for white matter registration and for surgical planning.

2 Background Theory

2.1 Diffusion Tensor Imaging and Tensor Similarity Measures

Diffusion tensor magnetic resonance imaging(DT-MRI) is a relatively new imaging modality that permits in vivo measures of the self-diffusion of water in living tissues. The tissue structure will affect the Brownian motion of the water molecules which will lead to an anisotropic diffusion. This anisotropic motion can be modelled by an anisotropic Gaussian, that can be parameterized by the diffusion tensor in each voxel [18] to create a 3D field of diffusion tensors.

The diffusion tensor is a 3×3 symmetric, semi-positive definite matrix. By diagonalizing the DT we obtain the eigenvalues ($\lambda_1, \lambda_2, \lambda_3$ where $\lambda_1 \geq \lambda_2 \geq \lambda_3$) and the corresponding eigenvectors ($\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$). Since the tensor is symmetric and semi-positive definite the eigenvalues will always be positive as long as they are unaffected by noise. The diffusion tensor can then be described in terms of its eigenvalues and eigenvectors.

$$D = (\mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3) \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} (\mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3)^T. \quad (1)$$

The largest eigenvalue and its corresponding eigenvector describes the quantity and direction of the principal diffusion.

Alexander et al.[19,20] have been exploring many similarity measures for tensors to perform elastic matching of diffusion tensor images. These measures do not only take the magnitudes of the diffusivity into account but also the directions. The most common similarity measure between two tensors are the tensor scalar product (TSP). This is a measure of the the overlap between two tensors:

$$D_1 : D_2 = \text{Trace}(D_1 D_2) = \sum_{j=1}^3 \sum_{i=1}^3 \lambda_{1i} \lambda_{2j} (e_{1i} e_{2j})^2. \quad (2)$$

The TSP is often normalized to avoid influence of the relative size of the two tensors. This will emphasize the shape and orientation of the tensor.

$$\text{NTSP}(D_1, D_2) = \frac{D_1 : D_2}{\text{Trace}(D_1) \text{Trace}(D_2)}. \quad (3)$$

This measure is dependent on the shape of the tensor and only a completely anisotropic tensor with diffusion in only one direction compared with itself will sum up to one. In some cases this can be a disadvantage but in our application this will be an advantage since it is the anisotropic regions that are of highest interest.

2.2 Geometrical Flows and Level Set Implementation

Geometrical flows and especially curvature- or curve shortening flows are becoming more and more important tools in computer vision. A curvature flow is a curve or surface that evolves at each point along the normal with the velocity depending on the curvature at that point. The theory is well developed for the two dimensional case and even though some of the properties of the 2D curves, such as the property of shrinking to a point under curvature flow, do not hold in the 3D case, the main part of the theories remains valid and works well for segmentation of 3D objects.

To use the geometrical flows for image segmentation, the evolution of the curve or surface has to depend on external properties dependent on the image features. A classical speed function to segment gray scale images is based on the gradient of the images and goes to zero when the surface approaches an edge.

A general flow for a 3D closed surface can be described as:

$$\frac{\partial S}{\partial t} = (F + \mathbf{H}) \vec{N}, \quad (4)$$

where F is an image based speed function and $\mathbf{H}$ is an intrinsic speed dependent on the curvature of the surface.

To solve this time dependent PDE we use the level set method, introduced by Osher and Sethian [1], where the evolving surface is considered as a constant level set of a function of a higher dimension. By doing this we obtain a numerically stable algorithm that easily handles topology changes of the evolving surface. In our case the function of higher dimension is the signed distance function, $\phi(t)$, of the evolving surface. This makes the evolution of the constant level set coincide with the evolution of $S(t)$. Thus, the evolution of the signed distance function is described by:

$$\phi_t = -(F + \mathbf{H}) |\nabla \phi|. \quad (5)$$

3 Method

3.1 Similarity Based Front Propagation

As mentioned in the introduction we propose a front propagation method that is based on the assumption that the diffusion is similar between two adjacent voxels within the same tract. To perform the segmentation a small initial surface is placed inside the tract we wish to segment and the surface is then propagated using the similarity measure in (3). The front propagates into a voxel with a speed proportional to the similarity between the diffusion tensor in the voxel and the diffusion tensors in the adjacent voxels lying inside the fiber. The front propagation speed is defined as:

$$F = \text{mean}(\text{NTSP}(D_i, D_{i-1}), \text{NTSP}(D_i, D_{i-2})) \quad (6)$$

where NTSP is the normalized tensor scalar product as in (3). D_i is the current voxel and D_{i-n} are the voxels found by following the normal to the surface n times backwards from the original voxel i , see Fig. 1. The flow does not necessarily evolve in the direction of the diffusion, but the shape of the diffusion tensor is not allowed to differ too much from the local neighborhood inside the fiber. This allows the surface to propagate towards the sides of the fiber tract and thereby segment the whole tract.

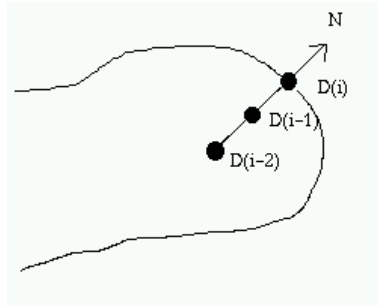


Fig. 1. Choice of adjacent voxels with respect to the normal.

3.2 Regularization

Due to a high level of noise in the DT-MRI a segmentation based on only the diffusive properties will be very irregular. To smooth the tracts while segmenting them we regularize the flow by adding a curvature dependent speed. Lorigo et al. introduced the use of a curvature definition from codimension 2 flows on surfaces with a thin, tubular structure [21]. Instead of using either mean curvature or gaussian curvature, which will normally destroy the tubular structure, they use the smaller principal curvature which is a combination of both curvatures. The smaller principle curvature, κ , is given by:

$$\kappa = H - \sqrt{H^2 - K},$$

where H is the mean curvature and K is the gaussian curvature. For the definitions of the mean- and gaussian curvature we refer to [22]. This definition of the curvature will smooth the tubes as if they were open curves in a 3D space, instead of smoothing their tubular form. We will use this definition for our curvature dependent smoothing term.

Our geometric flow now has the form:

$$\frac{\partial S}{\partial t} = (F + \kappa) \vec{N}. \quad (7)$$

This can easily be implemented with the level set method according to the above theories.

3.3 Thresholding

If the speed at one voxel is not equal to zero it will eventually lead to a propagation of the front at that voxel, even though the speed might be very small. To prevent unwanted propagation all speeds inferior to a certain threshold are set to zero. Thresholding is a very abrupt method so it risks to cause discontinuities

in the propagation. To avoid this, the Heaviside function defined in [23] is used to get a smoother thresholding.

$$H_{\epsilon}(x) = \begin{cases} 0 & \text{if } x < T - \epsilon \\ \frac{1}{2}\left[1 + \frac{x-T}{\epsilon} + \frac{1}{\pi} \sin(\pi(x-T)/\epsilon)\right] & \text{if } |x - T| \leq \epsilon \\ 1 & \text{if } x > T + \epsilon \end{cases} \quad (8)$$

where T is the selected threshold.

The surface evolution is stopped when the propagation speed has been sufficiently small for several succeeding iterations.

3.4 Implementation

The method has been implemented in Matlab 6.1 except for the reinitialization of the signed distance function, which has been implemented in C and compiled with the mex-library, so the function can be called from Matlab.

4 Validation and Results

4.1 Synthetic Tensor Fields

To test the method synthetic tensor fields have been created. Tensor values for one isotropic and one anisotropic tensor was taken from real data on DT-MRI of the brain of a healthy person. The isotropic tensor was then used as a background for synthetic fibers constructed of rotated anisotropic tensors.

With this method two different 3D tensor fields are constructed, presented in Fig. 2. The images show the largest eigenvector of the tensors at a cut along the z-axis. The first tensor field shows a semicircle to show the ability of following a fiber. The second tensor field simulates a branching fiber.

To make the tensor fields more realistic, noise is added [24]. This is done by making the inverse calculation to obtain the six amplitude images from which the diffusion tensors originally would have been acquired in DT-MRI and then add noise on the amplitude images. The added noise is an approximation the Rician noise [25], [26] as it would be on real MR data. After the noise has been added the tensor images are recreated. The resulting tensor fields can be seen in Fig. 2.

The method was then tested on the synthetic images with different levels of SNR. We have used a SNR of 8, 16 and 32. To start the segmentation a small initial surface is placed somewhere inside the synthetic fiber. Good results have been obtained with several different thresholds between 0.45 and 0.5. These threshold have then been used on the real MR data. Examples of resulting surfaces can be seen in Fig. 3, these are obtained with a SNR=8. Even though the synthetic tensor fields are very noisy the resulting surfaces are relatively smooth due our regularization that is performed as the surface is evolving.



Fig. 2. Synthetic fields: The principal directions of diffusion on a cut along the z-axis. Upper: Before noise is added. Lower: After noise is added.

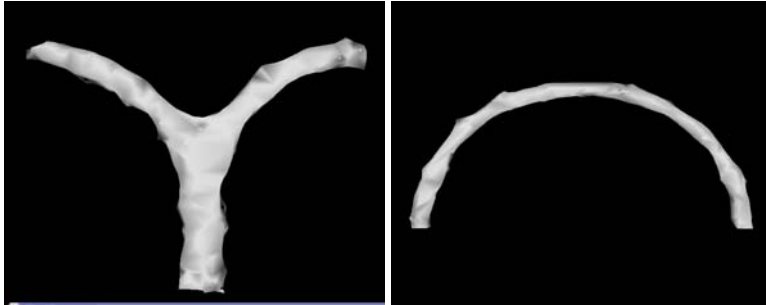


Fig. 3. Example of the resulting segmentation of synthetic vector field with SNR=8 and a threshold of 0.45. Similar results are obtained for thresholds between 0.45 to 0.5.

4.2 Real DT-MRI

MRI Data Acquisition. The diffusion tensor images we have used were acquired with a clinical MRI scanner (Magnetom Symphony; Siemens, Erlangen, Germany). The data was produced with a diffusion-weighted single-shot EPI sequence using the standard Siemens Diffusion Tensor Imaging Package for Symphony. We acquired 44 axial slices in a 128 by 128 matrix covering the whole brain of healthy volunteers, from the vertex to the end of the cerebellum. The

voxel size was 1.64 mm by 1.64 mm with a slice thickness of 3.00 mm without gap. Timing parameters were a TR of 1000 s and a TE of 89 s. Diffusion weighting was performed along 6 independent axes and we used a b-value of 1000s/mm² at a maximum gradient field of 30 mT/m. A normalizing image without diffusion weighting was also required. In order to increase the signal to noise ratio the measures were repeated 4 times. An anatomical T1-MP-Range was also performed during the same session. The whole examination lasted about one hour.

Preprocessing of Data. The preprocessing of the data and the geometric flow was carried out in Matlab 6.1. The diffusion tensor was computed for each voxel by linear combination of the log-ratio images according to Basser et al. [27]. The tensors were linearly interpolated component-wise between slices along the z-axis, to obtain a volume with a 3D regular grid of 1.64 mm.

To begin the segmentation an small initial surface is placed inside the fiber tract we wish to segment.

Results. The segmentation has been performed on three different DT-MR images. Two of the image acquisitions are from the same person. The results have been validated visually by comparing with post-mortem based neuroanatomical knowledge.

On the synthetic images we saw that several different thresholds have been possible for a good segmentation. On the real MR data the same range of thresholds have been used. Depending on the segmentation we desire the threshold has been varied within the predetermined range. In Fig. 4 the cortico spinal tract is segmented on the three different images. The threshold varies slightly dependent on the image acquisition. In Fig. 5 the corpus callosum has been segmented using two different thresholds. For a stricter threshold a smaller part is segmented but by choosing a lower threshold the surface passes further into the fibers and goes into some of the cortical association bundles.

The cortical association bundles can also be segmented separately as in Fig. 6. For every specific structure the threshold is a little different dependent on the shape of the tract and the anisotropic properties of the diffusion within the tract.

5 Discussion and Conclusion

We have presented a new method of segmenting entire fiber tracts by assuming that two adjacent voxels within the same tract has similar diffusion properties. The method manages to segment the larger tracts in the brain. This segmentation can be used as a base for future studies concerning for example quantification of the diffusion in the tracts or for white matter registration.

The segmentation results are sensitive for the choice of parameters. Since there is no objective measure of the exact solution on the brain images is it

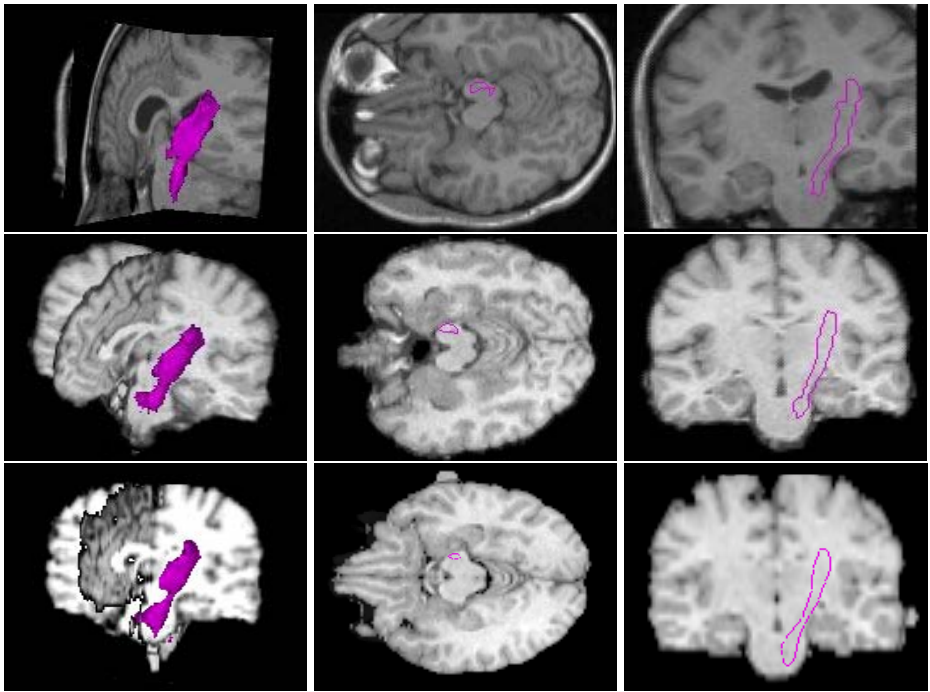


Fig. 4. Segmentation of the cortico spinal tract on three different brain images. The two last rows are images from the same patient but with two different image acquisitions. To obtain a similar segmentation for the three cases the threshold varies slightly dependent on the image acquisition. It is varied between 0.45 to 0.5.

difficult to determine exactly the optimal threshold. This is not necessarily a negative property, an advantage is the flexible segmentation when the choice of threshold determines how far to go in the fibers, see example in Fig. 5.

As mentioned in the introduction most of the existing methods are focusing on following the principal eigenvector of the diffusion tensor. The diffusion tensor contains a lot more information than just the main direction and magnitude of the flow. The other eigenvectors and eigenvalues also contain important data which is often ignored. Only looking at the principal direction also leads to a larger sensitivity to noise since a smaller deviation of the principal direction will lead to an important accumulative error. By exploring more of the tensor information we are creating flows that are less sensitive to noise. The similarity measures is based on the whole tensor and taking all eigenvectors into account.

An important advantage of our approach is the level set implementation. It provides an elegant tool for smoothing the segmented tracts and makes it possible to follow several paths simultaneously and effectively handle branchings and mergings of fibers.

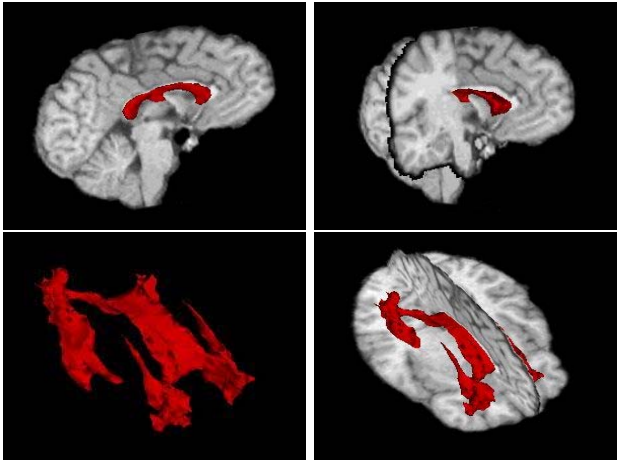


Fig. 5. Results for the similarity based flow on the first test brain. Segmentation of different parts of the corpus callosum for different thresholds. On the two upper images the threshold was set to 0.47 and for the two lower it was set to 0.45.

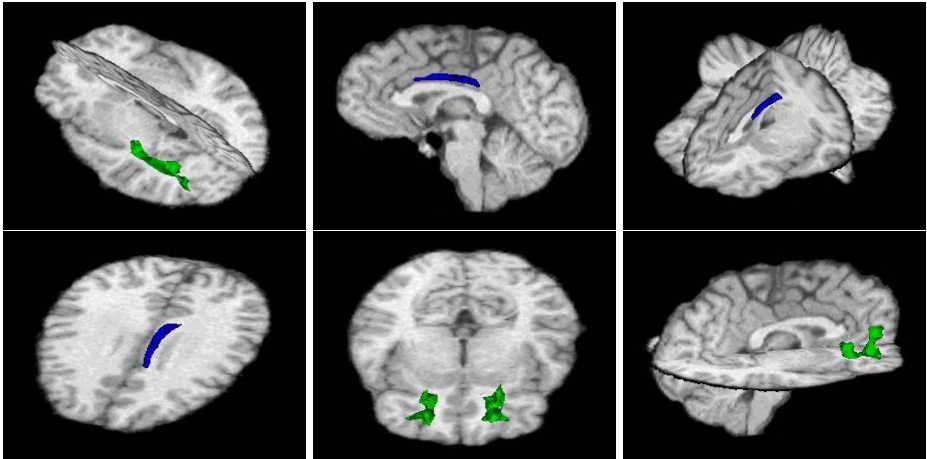


Fig. 6. Results for the similarity based flow on the first test brain. Segmentation of some of the cortical association bundles, threshold set at 0.45.

Another advantage of the flow is that we normalize with the total diffusion in each point. This will eliminate the influence of the diffusion strength and make a more correct measure of the common diffusion of the tensors. Calculating the NTSP with adjacent voxels lying inside the propagating surface leads to a regularization of the fiber tract in addition to the regularization performed with the curvature based propagation force.

References

1. S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.
2. M. Filippi, M. Cercignani, M. Inglese, M.A. Horsfield, and G. Comi. Diffusion tensor magnetic resonance imaging in multiple sclerosis. *Neurology*, 56(3):304–11, 2001.
3. J.A. Maldjian and R.I. Grossman. Future applications of dwi in ms. *J Neurol Sci*, 186(Suppl 1):S55–7, 2001.
4. S.E. Rose, F. Chen, J.B. Chalk, F.O. Zelaya, W.E. Strugnell, M. Benson, J. Semple, and D.M. Doddrell. Loss of connectivity in alzheimer's disease: an evaluation of white matter tract integrity with colour coded mr diffusion tensor imaging. *J Neurol Neurosurg Psychiatry*, 69(4):528–30, 2000.
5. M. Bozzali, A. Falini, M. Franceschi, M. Cercignani, M. Zuffi, G. Scotti, G. Comi, and M. Filippi. White matter damage in alzheimer's disease assessed in vivo using diffusion tensor magnetic resonance imaging. *J Neurol Neurosurg Psychiatry*, 72(6):742–6, 2002.
6. K.O. Lim, M. Hedehus, M. Moseley, A. de Crespigny, E.V. Sullivan, and A. Pfefferbaum. Compromised white matter tract integrity in schizophrenia inferred from diffusion tensor imaging. *Arch Gen Psychiatry*, 56(4):367–74, 1999.
7. J. Foong, M. Maier, C.A. Clark, G.J. Barker, D.H. Miller, and M.A. Ron. Neuropathological abnormalities of the corpus callosum in schizophrenia: a diffusion tensor imaging study. *J Neurol Neurosurg Psychiatry*, 68(2):242–4, 2000.
8. T. Klingberg, M. Hedehus, E. Temple, T. Salz, J.D. Gabrieli, M.E. Moseley, and R.A. Poldrack. Microstructure of temporo-parietal white matter as a basis for reading ability: evidence from diffusion tensor magnetic resonance imaging. *Neuron*, 25(2):493–500, 2000.
9. T.E. Conturo, N.F. Lori, T.S. Cull, E. Akbudak, A.Z. Snyder, J.S. Shimony, R.C. McKinstry, H. Burton, and M.E. Raichle. Tracking neuronal fiber pathways in the living human brain. *Proc Natl Acad Sci U S A*, 96(18):10422–7, 1999.
10. B.C. Vemuri, Rao M. Chen, Y., T. McGraw, Z. Wang, and T. Mareci. Fiber tract mapping from diffusion tensor mri. In *Proceedings of the IEEE Workshop on Variational and Level Set Methods in Computer Vision*, pages 81–88, 2001.
11. P.J. Basser, S. Pajevic, C. Pierpaoli, J. Duda, and A. Aldroubi. In vivo fiber tractography using dt-mri data. *Magn Reson Med*, 44(4):625–32, 2000.
12. P. Hagmann, J.P. Thiran, L. Jonasson, P. Vandergheynst, S. Clarke, and R. Meuli. Dti mapping of human brain connectivity: Statistical fibre tracking and virtual dissection. *NeuroImage*, in press, 2003.
13. G.J.M Parker, C.A.M. Wheeler-Kingshott, and G.J. Barker. Distributed anatomical brain connectivity derived from diffusion tensor imaging. In *IPMI 2001*, pages 106–120, 2001.

14. J. S. W. Campbell, K. Siddiqi, Baba C. Vemuri, and G. B. Pike. A geometric flow for white matter fibre tract reconstruction. In *International Symposium On Biomedical Imaging*, 2002.
15. P.G. Batchelor, F. Hill, F. Calamante, and D. Atkinson. Study of connectivity in the brain using the full diffusion tensor from mri. In *IPMI*, pages 121–133, 2001.
16. L. O'Donnell, Haker S., and Westin C.F. New approaches to estimation of white matter connectivity in diffusion tensor mri: Elliptic pdes and geodesics in a tensor-warped space. In *MICCAI2002*, 2002.
17. C.R. Tench, P.S. Morgan, M. Wilson, and L.D. Blumhardt. White matter mapping using diffusion tensor mri. *Magn Reson Med*, 47(5):967–72, 2002.
18. P.J. Basser, J. Mattiello, and D. Le Bihan. Mr diffusion tensor spectrography and imaging. *Biophys. J.*, 66:259–267, 1994.
19. D. Alexander, J. Gee, and R. Bajcsy. Similarity measures for matching diffusion tensor images. In *Proceedings BMCV'99*, 1999.
20. D. Alexander. Notes on indices of shapes and similarity for diffusion tensors. Technical report, 2001.
21. L.M. Lorigo, O. Faugeras, W.E.L. Grimson, R. Keriven, R. Kikinis, and C.F. Westin. Co-dimension 2 geodesic active contours for mra segmentation. In *IPMI 1999*, pages 126–139, 1999.
22. J.A. Sethian. *Level set methods and fast marching methods: Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*. 1999.
23. M. Sussman, E. Fatemi, P. Smereka, and S. Osher. An improved level set method for incompressible two-phase flows. *Computers and Fluids*, 27(5-6):663–680, 1998.
24. J.D. Tournier, F. Calamante, M.D. King, D.G. Gadian, and A. Connolly. Limitations and requirements of diffusion tensor fiber tracking: an assessment using simulations. *Magn Reson Med*, 47(4):701–8, 2002.
25. H. Gudbjartsson and S. Patz. The rician distribution of noisy mri data. *Magn Reson Med*, 34:910–914, 1995.
26. R.M. Henkelman. Measurement of signal intensities in the prescence of noise. *Med. Phys.*, 12:232–233, 1985.
27. P.J. Basser and C. Pierpaoli. A simplified method to measure the diffusion tensor from seven mr images. *Magn Reson Med*, 39:928–934, 1998.

Anisotropic Regularization of Posterior Probability Maps Using Vector Space Projections. Application to MRI Segmentation

M.A. Rodriguez-Florido¹, R. Cárdenes¹, C.-F. Westin², C. Alberola³, and
J. Ruiz-Alzola^{1,2}

¹ Centro de Tecnología Médica
Universidad de Las Palmas de Gran Canaria and
Hospital de Gran Canaria Dr. Negrín
Barranco de La Ballena s/n - 35020 - Canary Islands - Spain
{marf,ruben,jruiz}@ctm.ulpgc.es
<http://www.ctm.ulpgc.es>

² Laboratory of Mathematics in Imaging
Brigham and Women's Hospital,
Harvard Medical School - Boston (MA) - USA
westin@bwh.harvard.edu

³ Laboratorio de Procesado de Imagen
Universidad de Valladolid
Valladolid - Spain
caralb@yllera.tel.uva.es
<http://www.lpi.tel.uva.es>

Abstract. In this paper we address the problem of regularized data classification. To this extent we propose to regularize spatially the class-posterior probability maps, to be used by a MAP classification rule, by applying a non-iterative anisotropic filter to each of the class-posterior maps. Since the filter cannot guarantee that the smoothed maps preserve their probabilities meaning (i.e., probabilities must be in the range $[0, 1]$ and the class-probabilities must sum up to one), we project the smoothed maps onto a probability subspace. Promising results are presented for synthetic and real MRI datasets.

1 Introduction

Classical approaches to optimal statistical classification usually resort to a filtering step followed by the classifier itself. Ideally the filter should whiten the data in order to achieve optimality with rules classifying independently every voxel. The complexity of images, even more with medical images, makes it unfeasible to design practical whitening filters. Therefore, independent voxel classification leads to suboptimal results with scattered miss-classifications. Several approaches to regularize the resulting classification can be thought of. For example, just to name some of them, it is possible to prefilter the images, to post-regularize the

independent classification using relaxation labeling or mathematical morphology, or to use the more formal Markov Random Fields approach.

Teo et al. [13] have proposed a new segmentation technique where the class-conditional posterior probabilities maps obtained from the raw data, i.e. without any pre-filtering step, are smoothed using the well-known Perona-Malik diffusion scheme [8]. The classification is then carried out by using independent MAP (Maximum a Posteriori) rules on the smoothed posterior maps.

The results obtained with this technique are better than with other schemes because the regularizing diffusion produces piece-wise-constant posterior probability maps, which yield piece-wise “constant” MAP classifications. However, it presents some difficulties: Each posterior probability (a scalar field for every class) is smoothed independently by non-linear diffusion, ignoring the intrinsic relationship among them, and therefore not guaranteeing the posteriors to sum up to one for every voxel in each diffusion step. To resolve this, the authors normalize the addition of posterior probabilities after each discrete iteration, assuming that every class posterior probability is positive or zero.

Some extensions of this classification framework have been provided, specially in the fields of MRI (Magnetic Resonance Imaging) and SAR (Synthetic Aperture Radar). In particular, [12] has shown that the framework can be seen as a MAP solution of a discrete Markov Random Field with a non-interacting, analog discontinuity field, and [7] extended it by using a system of coupled PDE’s in order to guarantee for the diffusion process that the posterior probabilities are positive and sum up to one in each voxel.

In this work we propose an alternative approach to regularize the posterior probability fields. The major differences to previous schemes are:

1. Our approach is strictly anisotropic. Notice that the conventional Perona-Malik diffusion scheme is only homogeneous [15], since it uses a scalar function of the norm of the gradient as diffusivity.
2. It is not based on coupled PDE’s, but it uses anisotropic-adaptive multidimensional filtering on every class-posterior probability map. A local structure tensor, extracted from the input image, shapes the filter response adaptively, and the solution is obtained in a single step by linear combination of a basis filters outputs [3] [10].
3. In order for the filtered probabilities maps to sum up to one and be positive, a vector spaces projection approach POCS (Projection on Convex Sets) algorithm is proposed, i.e, the filtered maps are projected onto the bounded hyper-plane defined by this condition.

The paper is organized as follows: First, after this introduction to the problem, we review briefly the concepts of *regularization* and *vector spaces projections*, and their relationship. Second, we introduce the POCS approach to MAP segmentation. Then we review the multidimensional anisotropic filter used to smooth the posterior maps. And finally, we discuss our results with both synthetic and real datasets (MRI volumes), and propose some future lines of work.

2 Regularization and Vector Space Projections

There are several problems in image processing that are considered as *ill-posed* in the sense of Hadamard [1] and Tikhonov [14]: A problem is *ill-posed* if there is no guarantee of the solution existence, the solution is not unique, or the solution does not depend continuously on the input data (small variations of initial conditions of the problem lead to big variations on the solution). Typical examples of ill-posedness are: edge detection, visual interpolation, structure from stereo, shape from shading, computational of optical flow, and typically inverse problems.

Regularization theory provides a convenient way to solve ill-posed problems and to compute solutions that satisfy prescribed smoothness constraints. In fact, *regularization* can be seen as the restriction of the space of admissible solutions by introducing *a priori* input data information. As Poggio et al. [9] say in their classical paper, to solve an *ill-posed* problem expressed like $\mathbf{H}\mathbf{x} = \mathbf{y}$, is to find the value of $\mathbf{x}$ that minimize the expression:

$$\|\mathbf{H}\mathbf{x} - \mathbf{y}\|^2 + \lambda \|\mathbf{R}\mathbf{x}\|^2 \quad (1)$$

where λ is the regularization coefficient, $\mathbf{R}$ the regularization stabilizing operator, and $\mathbf{H}$ is a linear operator.

Basically, we can distinguish two ways to regularize the solution:

- Variational Regularization ([14]). An energy functional of the regularized solution $f_r(\mathbf{x})$ is minimized with respect to the input data $f_i(\mathbf{x})$:

$$E[f_r] = \frac{1}{2} \int ((f_i - f_r)^2 + \sum_{k=1}^{\infty} \lambda_k (\frac{\partial^k f_r}{\partial \mathbf{x}^k})^2) d\mathbf{x} \quad (2)$$

with $\lambda_k \geq 0$, such as if $\lambda_j = 0 \forall j > n$ the regularization is called *nth* order.

If the energy functional can be expressed with only sums of quadratic terms of the derivatives of the solution, it is equivalent to a linear filtering given by a smoothing operator.

- Statistical approach ([5]) that fixes the statistical properties of the solution space. The Bayes estimation taking the appropriate probability distributions is reduced to an expression similar to the regularization in the sense of Tikhonov, and if the *a priori* information is modeled in terms of a Markov Random Field (MRF) the maximum a posteriori estimation of the MRF is equivalent to a variational principle of the form of (2), as we mentioned in the introduction.

Signals with different properties form different subspaces in the general Hilbert signal space. These subspaces can be overlapped or disjointed, depending on the constraints. Finding the best approximation of a signal in a subspace can be achieved by projecting orthogonally the signal onto it. For example, the traditional sampling paradigm with ideal prefiltering could be seen as an orthogonal projection of the signals onto the subspace of band-limited signals.

Therefore, projections onto *vector spaces* are useful tools to find a subspace formed by signals that satisfy multiple constraints. Then, an interesting question arises: *Could the regularization be seen as a “projection” of the initial signal onto a subspace that satisfies the imposed regularity properties?* A partially answer to this question is possible if we consider the Sobolev spaces ¹, because *n*th order linear regularization is a mapping between Sobolev spaces, where the target space is included into the starting space: $\mathbf{R} : S_i \rightarrow S_r$ with $S_r \subset S_i$. However, the projection is a idempotent operator, and regularization can not be strictly called a projection. Hence, we can only give a partial affirmative answer to the previous question.

3 POCS-Based MAP Classification

The goal of any segmentation scheme is to label every voxel as belonging to one out of N possible classes $w_1 \dots w_N$. We will assume that a posterior probability model $p(w_i/z(\mathbf{x}))$ of the class w_i conditioned on the image intensity z is known for every voxel $\mathbf{x}$. Therefore there are N scalar maps, corresponding to the posterior probabilities of each class for every voxel, which can be stacked in a vector map of posterior probabilities. It is obvious that the components of the vector map must be all greater than or equal to zero and sum up to one, i.e., the L_1 -norm is equal to one for every voxel.

Conventional MAP proceeds by selecting the class with the maximum posterior probability, i.e. by choosing the greater component of the vector probability map at each voxel.

Now, let us consider that the vector probability map is regularized before any decision is made, in order to obtain a smooth behavior of the posterior probabilities inside regions and steep changes in their boundaries. To this extent the structure of the original image must be taken into account in order to obtain a structure tensor to steer the smoothing process: probabilities must be smoothed inside regions belonging to the same class and along their boundaries, but not across them. It is clear that the filtered vector map must also have all its components greater than or equal to zero and, for every voxel, the L_1 -norm must be equal to one. This constraint can be seen at every voxel as the hyper-plane $P_1 + \dots + P_N = 1$, where P_i corresponds to the i -th component of the posterior probability vector, i.e., to the posterior probability of the class i . See figure 1 for the three classes case.

This condition is not easy to meet for adaptive anisotropic smoothers in general. A first possibility to overcome this problem is to normalize the posterior vector map using a partition function, i.e, by dividing each vector component by the L_1 -norm at that voxel. This normalization does not change the result of the MAP rule though it becomes crucial if more than one iteration (smoothing and normalization) are carried out before the MAP rule is applied.

¹ A Sobolev space of order n is the space of all functions which are square integrable, and have derivatives well-defined and square integrable up to order n .

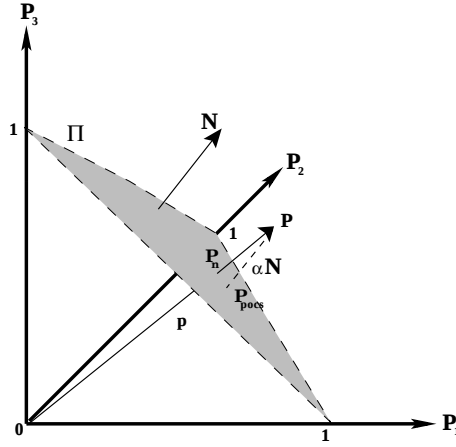


Fig. 1. An example of a three classes case with a posterior probability vector $\{P_1, P_2, P_3\}$. On the hyper-plane of interest Π , the points P_n and P_{pocs} result from the normalization of components after processing, and the filtered from POCS projection, respectively. $\mathbf{N}$ denotes the normal vector to the plane Π , $\mathbf{p}$ every filtered vector, and α a scalar factor.

Another alternative comes from projecting orthogonally the filtered class posteriors onto the constraint hyper-plane. This gives the closest class-posterior probabilities to the filtered ones that satisfy the constraint of summing up to one. To enforce that every component (every class-posterior probability) of the posterior probability vector being greater than or equal to zero, a new orthogonally projection must be done onto the hyper-plane restricted to the first hyper-quadrant. See figure 2 for the three classes case.

This technique is known as *POCS*, from *Projection Onto Convex Sets* [6]. POCS is a special case of the composite mapping algorithm that has been widely used in a variety of settings such as tomography and image restoration. The main idea of this technique consists of projecting the space of solutions onto a set attached to some *convex* constraints (the projection into the associated convex set is unique).

The projection of the filtered posterior vector onto the constraint hyper-plane can be easily obtained as follows:

$$\mathbf{P}_{\text{POCS}} = \mathbf{P} + \frac{1 - \langle \mathbf{P}, \mathbf{N} \rangle}{\|\mathbf{N}\|^2} \mathbf{N} \quad (3)$$

where $\mathbf{P}$ and $\mathbf{N}$ denote the posterior probability vector and the normal vector to the hyper-plane of interest, respectively.

4 The Smoothing Filter

The application of a smoothing operator is a well known technique of regularization, and as we mentioned before, this is a special case of the variational linear

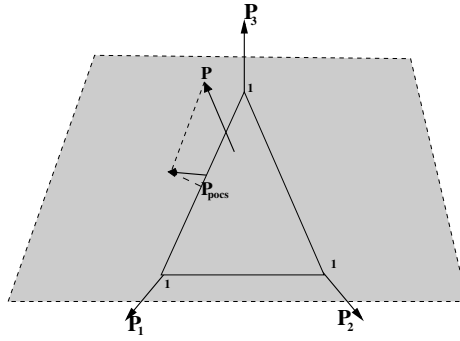


Fig. 2. Constraint of posterior probability greater than or equal to zero of the three classes case. The notation is the same than in figure 1. Note the second projection onto the line $P_3 + P_1 = 1$.

regularization approach. Traditionally, a PDE-based anisotropic diffusion filter is used in order to smooth the inside regions while preserving steep discontinuities across the borders. Here, we propose to use a non-iterative anisotropic filter which renders very good results.

A few words about *unsharp-masking*, a conventional enhance technique used by photographers to increase the importance of details in their pictures, will help us to introduce our anisotropic filter. Unsharp-masking is based on a decomposition of the image in a low pass component and a high-pass one. The high pass component is obtained from the difference between the image and the low pass component. If the weight of the high-pass is increased, the level of detail will be increased too. Then, we can consider that:

- The amount of detail could be adaptive at every point, weighting the high-pass component locally.
- The anisotropy of the structures should be taken in account to control in which orientations the high-pass component is important, and leaving the rest with a low pass component. This allows to smooth in the orientations of less structure (homogeneous regions, along edges, etc), preserving high structure elements.

Following these ideas, Knutsson et al. [4], introduced the local anisotropy information to process 2D grey-scale images. This was later generalized to N -dimensional images [3], developing an anisotropic N -dimensional adaptive filter, weighted by the local complexity of the dataset. Basically, this technique makes a linear combination of the output of a set of basis filters: one low pass (ideally a Wiener filter) and a set of high-pass filters in different orientations. The low pass filter fixes the scale of the dataset, and the high-pass ones the detail in each orientation. The contribution of each high basis component is related to the local dataset structure, and it is given by some coefficients. These coefficients should convey the importance of the information in every orientation with respect to the local structure, giving more weight to that orientation closest to the major direction of variation. Then, since in our approach the local complexity is coded

by the local structure tensor $\mathbf{T}_i$, [2], to obtain the importance of every orientation, we have to estimate the projection of the eigenvectors of the local structure tensor at every point onto every vector orientation $\hat{\mathbf{n}}_k$, i.e., the inner product of the normalized local structure tensor $\mathbf{C}$ and a basis tensor $\{\mathbf{M}^k\}$ associated to $\hat{\mathbf{n}}_k$. The basis $\{\mathbf{M}^k\}$ is the *dual basis* of $\mathbf{N}_k = \hat{\mathbf{n}}_k \hat{\mathbf{n}}_k^T$, unique and obtained by the biorthogonality relation:

$$\langle \mathbf{M}^k, \mathbf{N}_l \rangle = \delta_l^k \quad (4)$$

with δ_l^k the *Kronecker's delta*.

The minimum number of coefficients is the minimum number of high-pass filters, and it will be given by the relation between the data dimension N and the rank of this local structure tensor: $\frac{N(N+1)}{2}$. The process is described by the equation 5, where $S_{lp}(\mathbf{x})$ and $S_{hp_k}(\mathbf{x})$, denote the outputs of the low-pass and high-pass basis filters, respectively.

$$S_{AAF}(\mathbf{x}) = S_{lp}(\mathbf{x}) + \sum_{k=1}^{N(N+1)/2} \langle \mathbf{C}, \mathbf{M}^k \rangle S_{hp_k}(\mathbf{x}) \quad (5)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of tensors.

Figure 3 shows the flux diagram of our filtering approach is shown for a 2D grey-level image. The local structure is represented with an ellipsoid associated to the quadratic form of the structure tensor. The parameters used has been: $\rho_{lp} = \frac{\pi}{2}$ and $\eta = 2$ (a Gaussian filter).

In the Fourier domain, the basis filter $\{Lp, Hp_1, \dots, Hp_{N(N+1)/2}\}$ used, is given by the expression:

$$Lp(\boldsymbol{\nu}) = \begin{cases} e^{-(\rho^\eta \cdot \Delta)} & \text{if } 0 \leq \rho \leq \rho_{lp} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

$$Hp_k(\boldsymbol{\nu}) = (1 - Lp(\boldsymbol{\nu})) \cdot (\hat{\boldsymbol{\nu}}^T \hat{\mathbf{n}}_k)^2$$

with $\boldsymbol{\nu}$ the N -dimensional frequency vector, $\rho = \|\boldsymbol{\nu}\|$, ρ_{lp} the magnitude of the frequency where the Lp filter is $\frac{1}{2}$, and Δ and η filters shape parameters, related by: $\Delta = \frac{\ln 2}{\rho_{lp}^\eta}$.

A comparison of our filtering approach with an anisotropic diffusion technique, in the context of 3D medical imaging, can be found in [10].

5 Results

In order to illustrate the proposed approach we show its results on a simple synthetic image with two vertical regions (see Fig. 4.a). Each region is generated from IID Gaussian distributions with identical typical deviations (30) and different mean values (10 on the left region and 70 on the right one). Figure 4.b shows the results achieved with the optimal MAP classifier considering homogeneous priors (0.5 each). Figure 4.c shows the results achieved by applying the

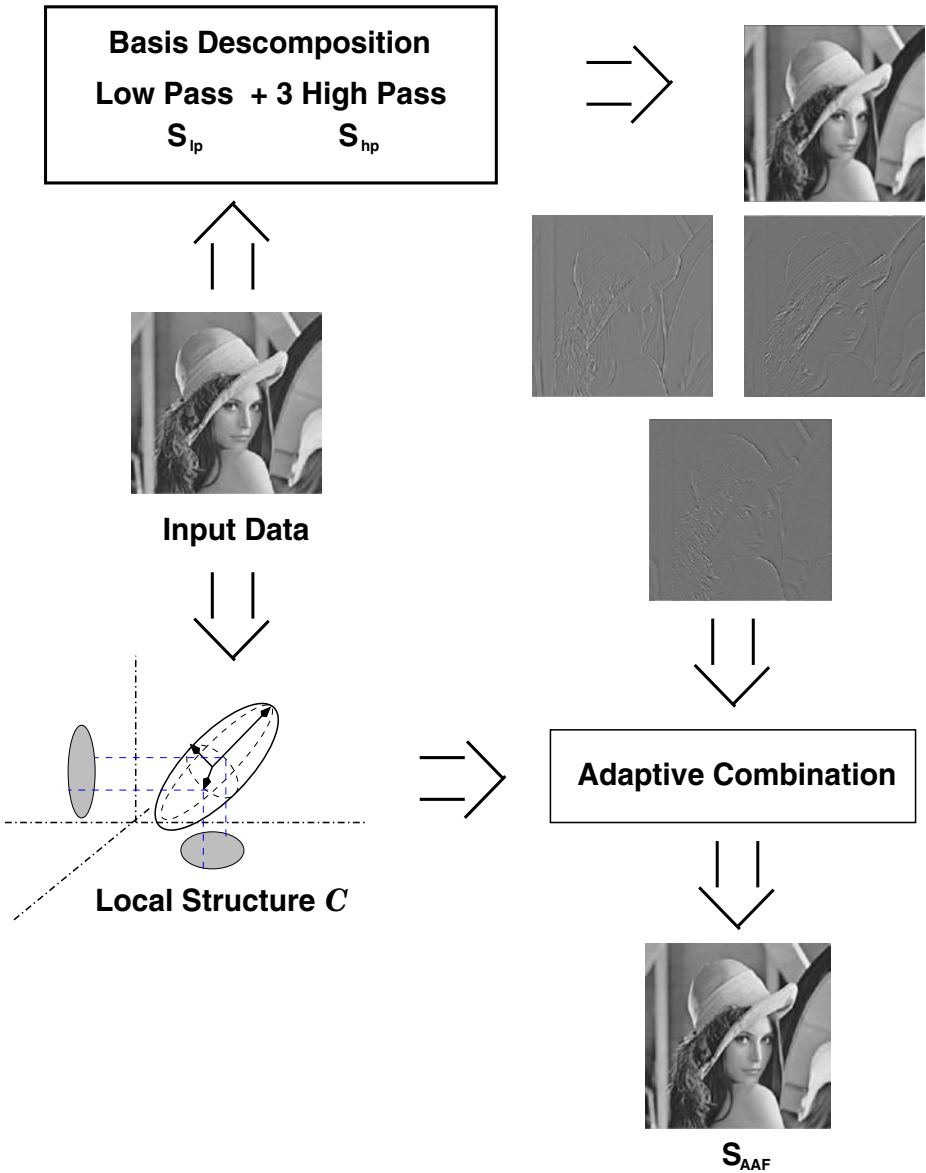


Fig. 3. Flux diagram of our filtering approach in a 2D scalar field.

MAP rule after the posteriors have been anisotropically smoothed and POCS-projected. Notice how inside each region the probability smoothing provides the correct classification and how the edge between both of them is almost optimally preserved.

To illustrate the performance of the method, in the case of real data, we apply it to the classification of the three main brain classes: gray matter (GM),

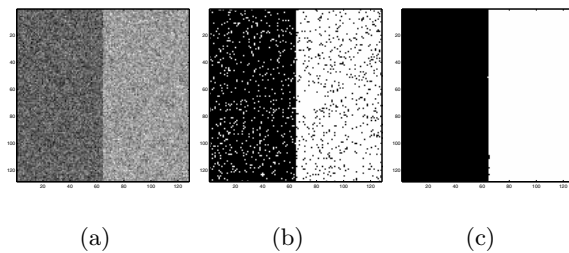


Fig. 4. **a)** Synthetic image with two regions **b)** MAP classification **c)** Regularized MAP classification

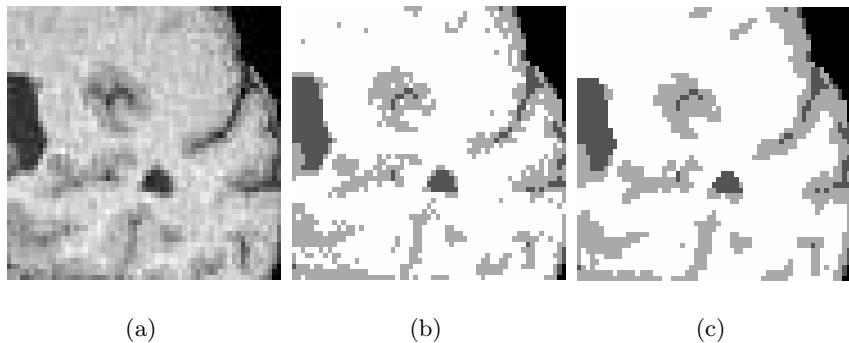


Fig. 5. **a)** Zoomed MRI brain image **b)** MAP classification **c)** Regularized MAP. Black: Background, Dark Gray: CSF, Gray: GM, and White: WM.

white matter (WM) and cerebrospinal fluid (CSF), from a 3D ($256 \times 256 \times 160$) MRI data. The brain is first extracted using a skull stripping automatic algorithm (thresholding followed by a light erosion, a hard opening, and region growing in order to obtain a brain mask that is applied to the original MRI data). Figure 5.a shows a zoomed area of a slice from the original MRI volume, Fig. 5.b shows the MAP classification (the parameters are provided from a ground-truth segmentation) and Fig. 5.c shows the regularized MAP segmentation using the same probabilistic characterization. The labels for the segmentation are white for WM, gray for GM, dark gray for CSF, and black for the background. Notice how the regularized MAP provides much more spatially coherent classifications while preserving the borders.

6 Discussion and Future Directions

In this paper we have proposed an approach to improve conventional independent voxel-wise MAP classifications. The approach is non-iterative and fast, and produces piecewise segmentations while preserving the class-borders. Our preliminary results are encouraging, though much more validation is still needed to

introduce it in routine clinical segmentations. Extensions of the approach include its use in arbitrary vector fields with different constraints. In fact, recently, we have developed a new approach to regularize tensor fields [11], using a variation of this technique.

Acknowledgments. The first author is funded by a FPU grant at the University of Las Palmas de Gran Canaria. This work is partially funded by the research project TIC2001-3808, by the Castilla y León Government project joint to the research grant VA91/01, and the US grants NIH P41-RR13218, and CIMIT.

References

1. Hadamard J. Lectures on the Cauchy Problem in Linear Partial Differential Equations. Yale University Press, 1923.
2. Knutsson, H. Representing local structure using tensors. In 6th Scandinavian Conference on Image Analysis. Oulu, Finland, pages 244–251, 1989.
3. Knutsson H., Haglund L., Bårman H., Granlund G. H. A framework for anisotropic adaptive filtering and analysis of image sequences and volumes. In Proceedings ICASSP-92, San Francisco, CA, USA, March 1992. IEEE.
4. Knutsson, H., Wilson, R. and Granlund, G.H. Anisotropic non-stationary image estimation and its applications-part I: Restoration of noisy images. IEEE Trans. on Communications. COM-31, 3:388–397, 1983.
5. Marroquin J.L. Probabilistic Solution of Inverse Problems. PhD thesis, M.I.T., Boston (MA) – USA, 1985.
6. Moon T.K. and Stirling W.C. Mathematical Methods and Algorithms for Signal Processing. Prentice-Hall, 2000.
7. Pardo A. and Sapiro G. Vector probability diffusion. IEEE Signal Processing Letters, 8(4):106–109, 2001.
8. Perona P., Malik J. Scale-Space and edge detection using anisotropic diffusion. IEEE Trans. on Pattern Analysis and Machine Intel., 12(7):629–639, July 1990.
9. Poggio T., Torre V. and Koch C. Computational Vision and Regularization Theory. Nature, 317:314–319, 1985.
10. Rodriguez-Flrido M.A., Krissian K., Ruiz-Alzola J., Westin C.-F. Comparison between two restoration techniques in the context of 3d medical imaging. Lecture Notes in Computer Science - Springer-Verlag Berlin Heidelberg, 2208:1031–1039, 2001.
11. Rodriguez-Flrido M.A., Westin C.-F., Castaño C. and Ruiz-Alzola J. DT-MRI Regularization Using Anisotropic Tensor Field Filtering. submitted to MICCAI03, 2003.
12. Teo P.C., Sapiro G., Wandell B.A. Anisotropic smoothing of posterior probabilities. In 1997 IEEE International Conference on Image Processing – October 26–29 – Washington, D.C., pages 675–678, 1997.
13. Teo P.C., Sapiro G., Wandell B.A. A method of creating connected representations of cortical gray matter for functional mri visualization. IEEE Trans. Medical Imaging, 16:852–863, 1997.
14. Tikhonov A., Arsenin V. Solutions of ill-posed problems. Wiley, New-York, 1977.
15. Weickert, J. Anisotropic Diffusion in image processing. Teubner-Verlag, Stuttgart, 1998.

Fast Entropy-Based Nonrigid Registration^{*}

Eduardo Suárez¹, Jose A. Santana¹, Eduardo Rovaris¹,
Carl-Fredrik Westin², and Juan Ruiz-Alzola^{1,2}

¹ Medical Technology Center,
Univ. Las Palmas of GC & Gran Canaria
Dr. Negrín Hospital, SPAIN
eduardo@ctm.ulpgc.es

² Laboratory of Mathematics in Imaging,
Brigham and Women's Hospital and
Harvard Medical School, USA

Abstract. Computer vision tasks such as learning, recognition, classification or segmentation applied to spatial data often requires spatial normalization of repeated features and structures. Spatial normalization, or in other words, image registration, is still a big hurdle for the image processing community. Its formulation often relies on the fact that correspondence is achieved when a similarity measure is maximized. This paper presents a novel similarity measuring technique based on a *coupling function* inside a template matching framework. It allows using any entropy-based similarity metric, which is crucial for registration using different acquisition devices. Results are presented using this technique on a multiresolution incremental scheme.

1 Introduction

Registration consists of finding the spatial correspondence between two coordinate systems with a scalar field defined on each one. For the two and three dimensional case, this is commonly called image registration. The correspondence is satisfied by means of a known similarity in the two datasets. This similarity is dependent on the application and is defined by high level information such as geodesic points and textures in satellite images, relevant anatomical points in medical images, or any other features for stereo matching or biosensing.

Datasets to be registered can therefore correspond to the same or to different subjects in the case of medical imaging or recognition, and can also be from the same or from different imaging modalities such as the different channels in satellite sensing. Putting into correspondence two images that can be topologically different (for example, in the case of a medical pathology) and where the pixel intensities measure different physical magnitudes (multimodality) poses a serious challenge that has sparked intensive research over the last years [1].

^{*} This work was supported by the spanish Ministry of Science and Technology and European Commission, co-funded grant TIC-2001-38008-C02-01, NIH grant P41-RR13218 and CIMIT, and spanish FPU grant AP98-52835909.

Particularly, in brain imaging, nonrigid registration is crucial for spatial normalization and brain understanding. Numerical approaches to nonrigid registration often rely on regularization theory, which separates the similarity measures of the image features from the smoothness constraints of the warping.

Voxel based registration methods can broadly be divided into two categories: methods based on a template matching technique, and those based on a variational approach. Template matching, also known as block matching in MPEG compression, finds the displacement for every voxel in a source image by maximizing a local similarity measure, obtained from a small neighborhood of the source image and a set of potential correspondent neighborhoods in a target image. The main disadvantages of its conventional formulation are that

- it estimates the displacement field independently in every voxel and thus spatial coherence is imposed to the solution
- it needs to test several discrete displacements to find a minimum and
- it has the inability of making a good match when no discriminant structure is available, such as in homogeneous regions, surfaces and edges (aperture problem).

Template matching was popular years ago due to its conceptual simplicity [2], but it does not impose any constraint on the resulting discrete fields, losing its place in this arena.

On the other hand, variational methods rely on the minimization of a functional (energy) that is usually formulated as the addition of two terms: data coupling and regularization, the former forcing the similarity between both datasets (target, and source deformed with the estimated field) to be high while the latter forcing the estimated field to fulfill some constraint (usually enforcing spatial coherence-smoothness).

Studies have already shown the power of the entropy-based similarity measures [3] to deal with multimodal registration. Their main drawback for nonrigid registration is the joint probability density function estimation, which must be known for every voxel displacement and computed on small neighborhoods of such voxels. There are two main approaches to overcome this problem. The first one [4], is to consider a global functional of a parameterized warping, and perform optimization on the parameters. It needs successive interpolation and global pdf estimation on every step, increasing the computational cost. The second approach is to compute local estimations of the gradient of the probability density function (hereinafter pdf) in order to steer local optimizations. This approach has been used in [5,6]. Nevertheless, they use a global pdf that is taken as static in gradient estimation.

In this paper we propose an entropy-based similarity measurement technique which can be used in a registration algorithm. It computes local entropy-based similarities, while using the appropriate probability density functions.

In order to show the efficiency of the matching technique a registration framework is needed. Section 2 introduces the registration framework exposed in [7] for later use. Section 3 describes the similarity measurement technique. Results are shown in Section 4 and conclusions in Section 5.

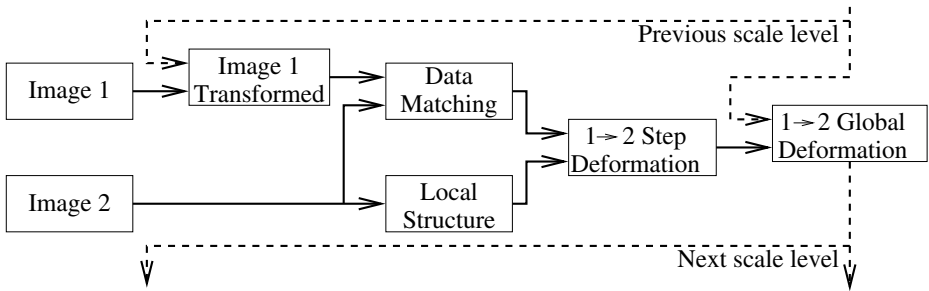


Fig. 1. Algorithm pipeline for one pyramidal level.

2 Nonrigid Registration

In order to introduce the entropy estimation scheme, it is first necessary to describe the registration framework where it is going to be used.

2.1 Multiscale

The registration algorithm consists of a pyramidal block-matching scheme, where the deformation field is regularized by weighting with local structure [7]. The algorithm works similarly to Kovačič and Bajcsy incremental multiresolution matching [8], which is based on a gaussian multiscale pyramidal representation. In the highest level, the deformation field is estimated by regularized template matching steered by the local structure of the image (details in sections below). In the next level, the source dataset is deformed with a deformation field obtained by spatial interpolation of the one obtained in the previous level. The deformed source and the target datasets on the current level are then registered to obtain the deformation field corresponding to the current level of resolution. This process is propagated to every level in the pyramid. The algorithm implementation is summarized in figure 1.

2.2 Data Matching

Template matching finds the displacement for every voxel in a source image by minimizing a local cost measure, obtained from a small neighborhood of the source image and a set of potential correspondent neighborhoods in a target image. The main disadvantage of template matching is that it estimates the displacement field independently in every voxel and no spatial coherence is imposed to the solution. Another disadvantage of template matching is that it needs to test several discrete displacements to find a minimum.

There exists some optimization-based template matching solutions that provide a real solution for every voxel, though they are slow [9]. Therefore, most template matching approaches render discrete displacement fields. Another problem associated to template matching is commonly denoted as the *aperture problem* in the computer vision literature [10]. This essentially consists of the inability of making a good match when no

discriminant structure is available, such as in homogeneous regions, surfaces and edges. When this fact is not taken into account the matching process is steered by noise and not by the local structure, since it is not available.

The registration framework keeps the simplicity of template matching while it addresses its drawbacks. Indeed the algorithm presented here consists of a weighted regularization of the template matching solution, where weights are obtained from the local structure, in order to render spatially coherent real deformation fields. Thanks to the multiscale nature of our approach only displacements of one voxel are necessary when matching the local neighborhoods.

2.3 Local Structure

Local structure measures the quantity of discriminant spatial information on every point of an image and it is crucial for template matching performance: the higher the local structure, the better the result obtained on that region with template matching.

In order to quantify local structure, a structure tensor is defined as $\mathbf{T}(\mathbf{x}) = (\nabla I(\mathbf{x}) \cdot \nabla I(\mathbf{x})^t)_\sigma$, where the subscript σ indicates a local smoothing. The structure tensor consists of a symmetric positive-semidefinite $D \times D$ matrix that can be associated to ellipses, i.e., eigenvectors and eigenvalues correspond to the ellipses axes directions and lengths respectively. D denotes the dimension of the space. A scalar measure of the local structure can be obtained as [11,12].

$$structure(\mathbf{x}) = \frac{(\det \mathbf{T}(\mathbf{x}))^{2/D}}{\text{trace } \mathbf{T}(\mathbf{x})}. \quad (1)$$

Small eigenvalues indicate the lack of gradient variation along the associated principal direction and, therefore, high structure would be represented by big rounded (no eigenvalue is small) ellipses. This way, anatomical landmarks will have the highest measure of local structure. Curves will be detected with lower intensity than points and surfaces will have even lower intensity. Homogeneous areas have almost no structure.

2.4 Spatial Regularization

The nature of the problem imposes the spatial mapping to be a diffeomorphism, therefore the Jacobian of the deformation field must be positive. This condition is preserved while regularizing small deformation fields. Large deformation fields with no spatial coherence would need a very destructive regularization to achieve invertibility. Thus, only one pixel displacements are allowed for matching. For every level of the pyramid the mapping is obtained by composing the transformation on the higher level with the one on the current level, in order to preserve the Jacobian positiveness condition.

Regularization is achieved by means of the *Normalized Convolution* [13], a refinement of weighted-least squares that explicitly deals with the so-called signal/certainty philosophy. Essentially the scalar measure of structure (the certainty) is incorporated as a weighting function in a least squares fashion. The field (the signal) obtained from template matching is then projected onto a vector space described by a non-orthogonal

basis, i.e., the dot products between the field and every element of the basis provide covariant components that must be converted into contravariant by an appropriate metric tensor. Normalized convolution provides a simple and efficient implementation of this operation.

Moreover, an applicability function is enforced on the basis elements in order to guarantee a proper localization and to avoid high frequency artifacts. This essentially corresponds weighting each basis element with the applicability function. In this application a Gaussian function will be used. Convolution with a Gaussian window can be implemented in a fast and efficient way, because it is a separable kernel.

In the three-dimensional case, we have used a basis consisting in three elements, which can be written as:

$$\mathbf{b}_1 \begin{cases} d_1(\mathbf{x}) = 1 \\ d_2(\mathbf{x}) = 0 \\ d_3(\mathbf{x}) = 0 \end{cases} \quad \mathbf{b}_2 \begin{cases} d_1(\mathbf{x}) = 0 \\ d_2(\mathbf{x}) = 1 \\ d_3(\mathbf{x}) = 0 \end{cases} \quad \mathbf{b}_3 \begin{cases} d_1(\mathbf{x}) = 0 \\ d_2(\mathbf{x}) = 0 \\ d_3(\mathbf{x}) = 1 \end{cases} \quad (2)$$

This way, we decouple the components of the displacement field and regularization is done applying independently normalized convolution to each component.

3 Entropy Estimation

In [7], the registration framework was tested using square blocks that were matched using the sum of squared differences and correlation coefficient as similarity measures. In this work we introduce entropy-based similarity measures into this framework, though it can be used by any algorithm based on template matching.

A similarity measure can be interpreted as a function defined on the joint probability space of two random variables to be matched. In the case of block matching, each block represents a set of samples from each random variable.

When this probability density function is known, mutual information can be computed as

$$MI(I_1, I_2) = \int_{\Omega} p(i_1, i_2) \log \frac{p(i_1, i_2)}{p(i_1)p(i_2)} di_1 di_2 \quad (3)$$

where I_1, I_2 are the images to register and Ω is the joint probability function space.

A discrete approximation is to compute the mutual information from the pdf and a small number N of samples $(i_1[k], i_2[k])$:

$$MI(I_1, I_2) \simeq \sum_{k=1}^N \log \frac{p(i_1[k], i_2[k])}{p(i_1[k])p(i_2[k])} = \sum_{k=1}^N f_p(i_1[k], i_2[k]), \quad (4)$$

where f_p is a *coupling function* defined on Ω .

Therefore, the local evaluation of the mutual information for a displaced block containing N voxels can be computed just by summing the coupling function f_p on the k samples that belong to this block.

We propose to compute a set of multidimensional images, each of them containing at each voxel the local similarity measure corresponding to a single displacement applied

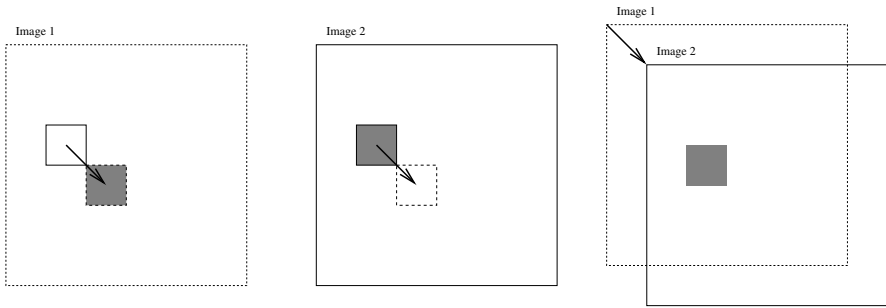


Fig. 2. *Left:* Target image to be matched; *Center:* Reference image where similarity measure is going to be estimated for every discrete displacement; *Right:* For every discrete displacement the similarity measure is computed for every voxel by performing a convolution.

to the whole target image. A decision will be made for each voxel depending on which displacement renders the biggest similarity.

A problem associated with local entropy-based similarity measures is the local estimation of the joint pdf of both blocks, since never enough samples are available. We propose to overcome this problem by using the joint pdf corresponding to the whole displaced source image and the target one.

The pdf to be used for a given displacement will be the global joint intensity histogram of the reference image with the displaced target image. This is crucial for higher pyramidal levels, where one voxel displacement changes drastically the pdf estimation.

It is straightforward to compute the local mutual information for a given discrete displacement in the whole image. This requires only the convolution of a square kernel representing the block window and the evaluation of the coupling function for every pair of voxels. Furthermore, since the registration framework only needs discrete deformation fields, no interpolation is needed in this step. Any similarity measure which can be computed as a kernel convolution can be implemented this way. A small sketch of this technique is shown in figure 2. For smoothness and locality reasons, we have chosen to convolve using gaussian kernels instead of square ones.

In order to achieve a further computational saving, equation 4 can be written as:

$$MI(I_1, I_2) \simeq \sum_{k=1}^N (\log p(i_1[k], i_2[k]) - \log p(i_1[k]) - \log p(i_2[k])) \quad (5)$$

Since the source image remains static, the term $\log p(i_1[k])$ is the same across the whole set of images with the local mutual information for each displacement. Therefore, this term does not affect to the decision rule, and can be disregarded, hence, avoiding to estimate the marginal pdf for the target image.

The displacement field tells about the displacement of a voxel in the source image. The similarity measure will be referred to the source image reference system (image 1). For a given voxel in the source image, the comparison of equation 5 for different displacement will always contain the same terms depending on $p(i_1[k])$. So that, we can take this term off and modify accordingly the coupling function to save computation cost.

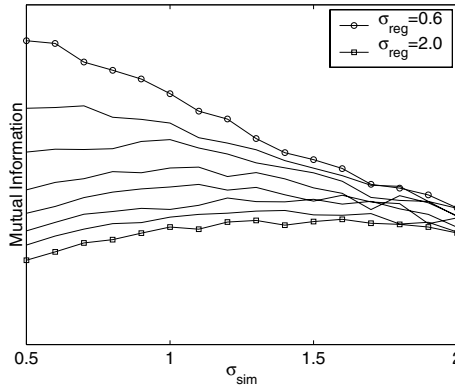


Fig. 3. Mutual Information as a function of the standard deviations used for similarity and regularization kernels.

Any other entropy-based similarity measure can be estimated in a similar way. The computational cost is then very similar to any other similarity measure not based on entropy

4 Results

Two experiments have been carried out. The former will give us quantitative results about the registration algorithm, while the latter will show us qualitative ones. Mutual information is the selected similarity measure used for next registration results. Datasets have previously been rigidly aligned and resampled.

In the first experiment, two T1-weighted images corresponding to different patients have been registered. The datasets have $90 \times 98 \times 72$ voxels. Seven multiscale pyramidal levels have been used.

To carry out the experiment, two kernel standard deviations σ_{sim} and σ_{reg} are needed. The former will be used for the convolution kernel of the coupling function sum in equation 4. The latter is for the kernel used in the regularization of the displacement field, that is, the normalized convolution applicability. Figure 3 shows the mutual information of the registered images as a function of σ_{sim} and σ_{reg} .

In a second experiment, an interpatient nonrigid registration has been done with two T1-weighted images of size 256×256 , with $\sigma_{reg} = 1.2$ and $\sigma_{sim} = 1.2$ and seven pyramidal levels. Registration is shown in figure 4. We can see the source image deformed onto the target image.

5 Conclusions and Future Work

Figure 3 shows that we get higher values of mutual information when using small σ_{reg} and σ_{sim} . However, by visual inspection, we have checked that the deformation is not smooth at all, and hence results are not valid using these values. As we move to very high σ_{reg} values, the regularization is so strong that it makes the deformation approach zero, making similarity drop to the similarity of the unregistered images.

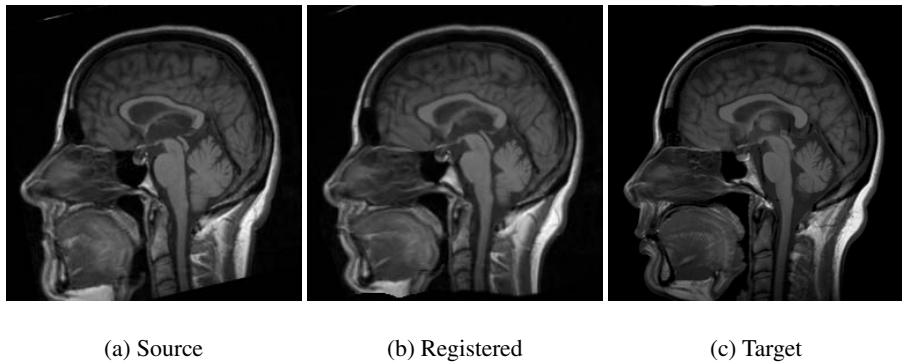


Fig. 4. Interpatient registration.

For middle values of σ_{reg} we find a reasonable behaviour in the local estimation of the similarity measure, that is, for high σ_{sim} the registration is worse because the measure is not local anymore, and for low values, there are too few samples to have a good estimation of the mutual information. We conclude that local estimation of mutual information has been done successfully with this technique. Visual inspection of the registration in figure 3 confirms this conclusion.

Despite this technique can be applied for intersubject multimodal registration, local variation of intensities may be present in one modality while not in the other, making the registration process fail. So then, its optimum application is for monomodal image registration. There, different scanning parameters, bias, or anatomical differences may occur and classical monomodal measures are not appropriate.

As future work, speedup of the algorithm will be achieved by means of a full C implementation. At the moment, the algorithm is implemented in Matlab and it takes about fifteen minutes for every registration described in experiment 1.

References

1. J.B.A. Maintz and M.A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, April 1998.
2. Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.
3. M. Holden, D.L.G. Hill, E.R.E. Denton, J.M. Jarosz, T.C.S. Cox, J. Goody, T. Rohlfing, and D.J. Hawkes. Voxel similarity measures for 3D serial MR image registration. *IEEE Trans. Med. Imag.*, 19(2):94–102, 2000.
4. D. Rueckert, L. I. Sonoda, E.R.E. Denton, S. Rankin, C. Hayes, D. L. G. Hill, M. Leach, and D. J. Hawkes. Comparison and evaluation of rigid and non-rigid registration of breast MR images. In *SPIE Med. Im. 1999: Im. Proc.*, pages 78–88, San Diego, CA, February 1999.
5. N. Hata, T. Dohi, S. K. Warfield, W. M. Wells, R. Kikinis, and F. A. Jolesz. Multimodality deformable registration of pre- and intraoperative images for MRI-guided brain surgery. In *MICCAI '98*, pages 1067–1074, 1998.
6. G. Hermosillo, C. Chef'd'Hotel, and O. Faugeras. A variational approach to multi-modal image matching. Technical Report 4117, INRIA, 2001.

7. Eduardo Suárez, C.-F. Westin, E. Rovaris, and Juan Ruiz-Alzola. Nonrigid registration using regularized matching weighted by local structure. In *MICCAI '02*, number 2 in Lecture Notes in Computer Science, pages 581–589, Tokyo, Japan, September 2002.
8. Stanislav Kovacic and R.K. Bajcsy. *Brain Warping*, chapter Multiscale/Multiresolution Representations, pages 45–65. Academic Press, 1999.
9. Eduardo Suárez, Rubén Cárdenes, Carlos Alberola, Carl-Fredrik Westin, and Juan Ruiz-Alzola. A general approach to nonrigid registration: Decoupled optimization. In *23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '01)*. IEEE Engineering in Medicine and Biology Society, october 2001.
10. T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:314–319, September 1985.
11. K. Rohr. Differential operators for detecting point landmarks. *Image and Vision Computing*, 15:219–233, 1997.
12. J. Ruiz-Alzola, R. Kikinis, and C.-F. Westin. Detection of point landmarks in multidimensional tensor data. *Signal Processing*, 81:2243–2247, 2001.
13. C.-F. Westin. *A Tensor Framework for Multidimensional Signal Processing*. PhD thesis, Linköping University, Sweden, SE-581 83 Linköping, Sweden, 1994. Dissertation No 348, ISBN 91-7871-421-4.
14. Jean-Philippe Thiran and Torsten Butz. Fast non-rigid resistration and model-based segmentation of 3d images using mutual information. In *SPIE*, volume 3979, pages 1504–1515, San Diego, USA, 2000.
15. P. Thévenaz and M. Unser. Optmization of mutual information for multiresolution image registration. *IEEE Transactions on Image Processing*, 9(12):2083–2099, 2000.
16. F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medical Imaging*, 16:187–198, 1997.
17. J.P.W. Pluim, J.B.A. Maintz, and M.A. Viergever. A multiscale approach to mutual information matching. In K.M. Hanson, editor, *Medical Imaging: Image Processing*, volume 3338 of *Proc. SPIE*, pages 1334–1344, Bellingham, WA, 1998. SPIE Press.
18. J. B. Antoine Maintz, Erik H. W. Meijering, and Max A. Viergever. General multimodal elastic registration based on mutual information. In K. M. Hanson, editor, *Medical Imaging 1998: Image Processing*, volume 3338 of *Proceedings of SPIE*, pages 144–154, Bellingham, WA, 1998. The International Society for Optical Engineering.
19. J.P.W. Pluim, J.B.A. Maintz, and M.A. Viergever. Image registration by maximization of combined mutual information and gradient information. *IEEE Transactions on Medical Imaging*, 19(8):809–814, 2000.
20. William M. Wells, Paul Viola, Hideki Atsumi, Shin Nakajima, and Ron Kikinis. Multi-modal volume registration by maximization of mutual information. *Medical Image Analysis*, 1:35–52, 1996.
21. Paul Viola and Williams M. Wells. Alignment by maximization of mutual information. *International Journal of Computer Vision*, 24:137–154, 1997.
22. C. Studholme, D.L.G. Hill, and D.J. Hawkes. Automated 3d registration of mr and ct images of the head. *Medical Image Analysis*, 1(2):163–175, 1996.

3D Reconstruction from a Vascular Tree Model

Luis Álvarez, Karina Baños, Carmelo Cuenca,
Julio Esclarín, and Javier Sánchez

Computer Science Department
Las Palmas University,
Las Palmas 35017 SPAIN

{lalvarez,kbaños,ccuenca,jesclarin,jsanchez}@dis.ulpgc.es
<http://serdis.dis.ulpgc.es/~{ }lalvarez/ami/index.html>

Abstract. In this paper, we present a vascular tree model made with synthetic materials and which allows us to obtain images to make a 3D reconstruction. We have used PVC tubes of several diameters and lengths that will let us evaluate the accuracy of our 3D reconstruction. In order to calibrate the camera we have used a corner detector. Also we have used Optical Flow techniques to follow the points through the images going and going back. We describe two general techniques to extract a sequence of corresponding points from multiple views of an object. The resulting sequence of points will be used later to reconstruct a set of 3D points representing the object surfaces on the scene. We have made the 3D reconstruction choosing by chance a couple of images and we have calculated the projection error. After several repetitions, we have found the best 3D location for the point.

1 Introduction

Given the difficulties to obtain medical images in good conditions, due to their privacy and diverse technical problems that these can present, such as: artifacts, occlusions, poor definition of the image, etc; we have developed a model of vascular tree using PVC.

This way we can obtain a series of images following the same technology of a rotational angiography, planning an arch of fixed radius about the model and obtaining the images with an angular separation of about 3 degrees. Once obtained, they were manipulated in order that the final result is as similar as possible to the angiographies, but without the problems previously mentioned. Besides, by having the 3D model it is possible to test the quality of the results obtained, knowing exact values such as: distances between bifurcations, the diameters of the glasses, etc; which will allow us to know the kindness of our results.

2 Obtaining the Images

We want to reconstruct images to apply in the Rotational Angiography field that it has the next features:

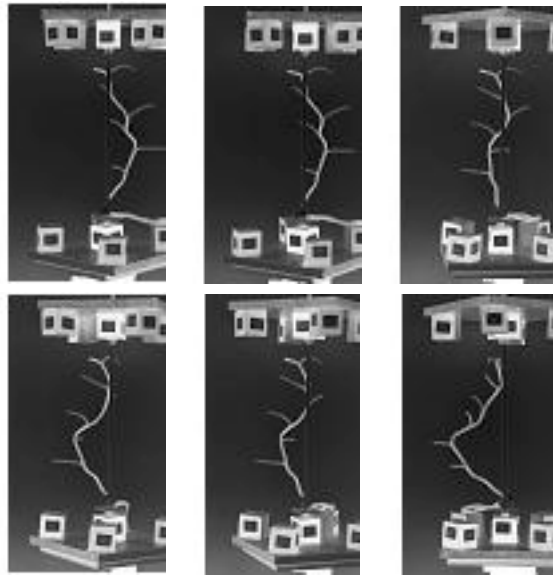


Fig. 1. Acquired images.

The series of images in Rotational Angiography is acquired while the imaging assembly rotates in a continuous arc around the patient. The whole acquisition is rather fast, so that the complete series can be acquired with a single injection of contrast agent.

The quality of the individual images is generally fully adequate for diagnosis, with the following added advantages: wide range of projections, optimum views of vascular structures.

For the 3D reconstruction, it is essential for the images to precisely match each other. This requires an extremely stable and reproducible image geometry. The system is calibrated to compensate for distortion in the image intensifier such as pincushion and the varying distortion caused by movement through the magnetic field of the earth.

The images are acquired in the rotational angiography mode over an angle of 180 degrees. The run maybe carried out in one of three different angulations: -30 degrees cranial, 0 degrees axial, 30 degrees caudal. Images are acquired at a frames rate of 12.5 frames by seconds, and a rotation speed of up to 30 degrees per second, the whole acquisition takes 8 seconds resulting in an average of 100 images per run.

We have made the images with an angle of 3.6 degrees and approximately three meters of distance from the model using a of 70 mm focal distance with a digital camera and because the characteristics of its digitizer then we have a focal distance 105 mm with an arc of 270 degrees that allows around 70 images. We have also obtained images of a calibrator in order to obtain the intrinsic parameters of the camera.

3 Multiscale Analisis and Calibration Camera

One of the main concepts of vision theory and image analysis is multiscale analysis. A Multiscale Analysis T_t associates, with an original image $u(0) = u_0$ a sequence of smoothed images $u(t, x, y)$ which depend upon an abstract parameter $t > 0$, the scale.

$$\begin{aligned} f(x, y) &\longrightarrow u(t, x, y) \\ u(0, x, y) &= f(x, y) \end{aligned}$$

The datum of $u_0(x, y)$ is not absolute in perception theory, but can be considered as the element of an equivalence class. If A is any affine map to the plane, $u_0(x, y)$ and $u_0(A(x, y))$ can be assumed equivalent from a perceptual point of view. Last but not least, the observation of $u_0(x, y)$ does not generally give any reliable information about the number of photons sent by any visible place to the optical sensor. Therefore, the equivalence class in consideration will be $g(u_0(A(x, y)))$, where $g(x)$ stands for any contrast function depending on the sensor. These considerations lead us to focus on the only multiscale analyses which satisfies these invariance requirements : The Affine Morphological Scale Space (AMSS). This multiscale analyses can be defined by a simple Partial Differential Equation:

$$u_t = t^{\frac{1}{3}}(u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy})^{\frac{1}{3}}$$

where $u(t, x, y)$ denotes the image analyzed at the scale t and the point (x, y) .

In order to calibrate a camera system we need to corner detection and this is very sensitive to noise. The AMSS multiscale analysis present the advantage that we know, analytically, the displacement of the corner location across the scales. Then we can search it in at the scale $t_n = t_0 + n\Delta t$, for $n = 1, \dots, N$, where u_t represents the discretization step for the scale and t_0 represents the initial scale that we use to begin to look for corners.

We compute for the scale t_0 the location of the extreme of the curvature that we denote by (x_0^i, y_0^i) , for $i = 1, \dots, M$, these points represent the initial candidates to be corners. We follow across the location (x_n^i, y_n^i) of the curvature extreme.

For each sequence (x_n^i, y_n^i) $n = 1, \dots, N$, we compute in a robust way (using orthogonal regression and eliminating outliers) the best line which fit the sequence of points, this line corresponds to the bisector line of the corner, and we can represent it as a straight line which equation:

$$(x(t), y(t)) = (x_0, y_0) + \tan\left(\frac{\alpha}{2}\right)^{-\frac{1}{2}} t (b_x, b_y)$$

where α is the angle of the corner and $\overrightarrow{b} = (b_x, b_y)$, is the unit vector in the direction of the bisector line of the corner, and t is the scale. Then we can find the corner doing $t = 0$ in this equation.

In order to calibrate the cameras system, we extract the characteristics of the sequence of views with a morphologic corner detector. This detector gives us

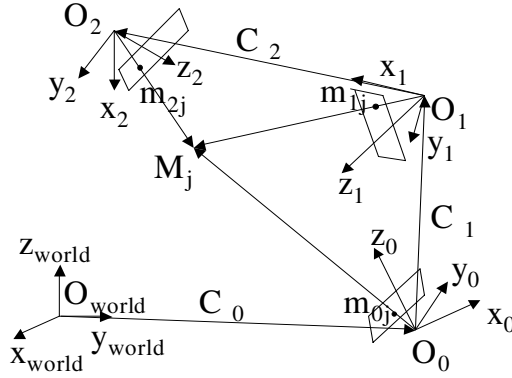


Fig. 2. Motion parameters derived from point matches.

sub pixel information. When the views are taken from very close positions, the conventional methods of calibration can be unstable, to solve this problem we divide the sequences of views into several sub sequences (in this way the optical centre displacements are bigger). Now, we calibrate every subsequence of view in an independent way.

In the last step, we make the calibration between the different sub sequences to obtain only one calibration. The method we have used is very stable, even when there are noise and small displacements between the optical centres.

4 Optical Flow

Once we have calibrated the camera system we propose a method for the recovering of disparity maps between pair of stereoscopic images. Disparity maps are obtained through a matching process in where we have to find for the pixels in the left image their correspondent on the right image. There are some methods, like correlation-based techniques, that estimate good matching points but do not generate smooth disparity maps for the whole image, so the solutions in this case are not continuous.

To improve the accuracy of the matching process we make use of the so-called epipolar geometry. This geometry represents the relation that exists between stereoscopic images. Thanks to this geometry, the method is able to look for correspondences in straight lines only.

The method we propose for the computation of disparity maps is based on an en-ergy minimization approach:

$$E(\lambda) = \int (I(\vec{x}) - I'(\vec{x} + \vec{h}(\lambda)))^2 + \int \nabla \lambda D(\nabla I) \nabla \lambda$$

Where I and I' are the stereoscopic images, λ is a parameter that gives us the distance between the point that it is the projection of m over the it epipolar

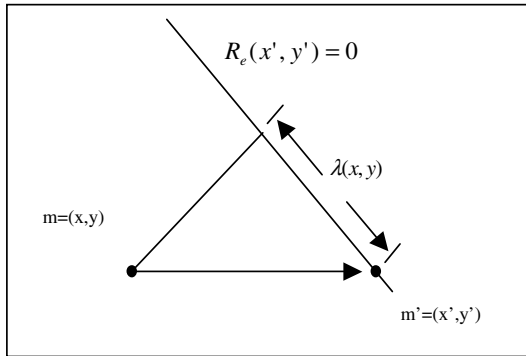


Fig. 3. Displacement function is parameterized in order to take advantage of the information given by the epipolar geometry.

straight line and the responsive point of m in I' , m' see image 3. The second term in the equation is used to regularize. D is a diffusion tensor that diffuses in one or another way depending where the point is placed. If gradient of I is high then the regularization is over the contour line and if it is low, we make regularization.

This energy consists of an attachment term that enables the process to find similar pixels in both images and a regularization term that is necessary to constraint the number of possible solutions and to generate smooth solutions. This method is a dense method in the sense that for every pixel on one image we obtain its correspondent on the other image.

When we minimize this energy, we obtain the Euler-Lagrange equations, which are represented by means of partial differential equations. This is a diffusion-reaction equation that behaves anisotropically at contours with high values for the gradient of the images and isotropically at homogeneous regions where the image gradient is low. The diffusion part is formulated in such a way that the discontinuities of the images are preserved. We use a scale-space and pyramidal strategy to allow the method to locate large displacements. Thanks to this energy minimization approach the resulting disparity maps that we may obtain are smooth by regions.

5 Corresponding Points and 3D Reconstruction

To search the corresponding points in every image, we start with a point in one image, and using the optical flow techniques, we search the corresponding point in the next image. Once this point is obtained, we go back and we search if the corresponding point in the first image is the start point. If it is true, we continue searching other point in the next image. Once we have this new point we come back until the first image verifying that the points calculates are the same points that we have with a small error. We finish this process when we search a point bad placed.

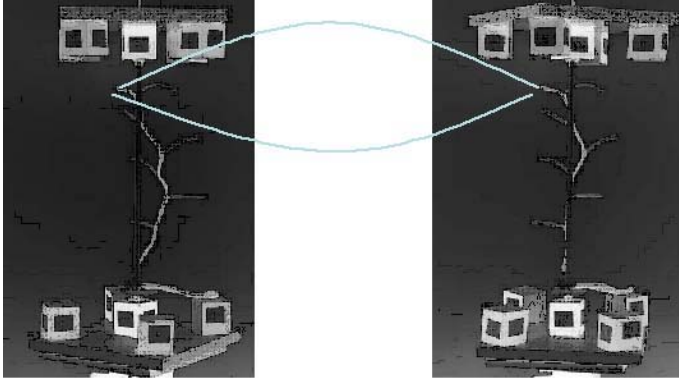


Fig. 4. Backward and Forward Optical Flow for a point in a couple of views.

First, we will describe an efficient algorithm for computing sequences of corresponding points with maximum lengths. Second, we will describe a fast algorithm for computing sequences of corresponding points, but in this case without maximum lengths. Experimental results show the validity of this algorithm. Both algorithms try to include a candidate to a sequence of n corresponding points to obtain a sequence of $n + 1$ points. We will explain different quality criterions, which are applicable to decide whether to add a new point to the sequence, or not.

In order to calculate the sequence of points in correspondence, we must know the Optical Flow from every view j to the precedent view $j - 1$ (backward), that we denote by $h_-^j(x) = h_-^j(x, y) = (u_-^j(x, y), v_-^j(x, y))^T$ and the Optical Flow from the view j to the next view $j + 1$ (forward), $h_+^j(x) = h_+^j(x, y) = (u_+^j(x, y), v_+^j(x, y))^T$.

The main idea is using the Optical Flow $h_+^j(x)$ to localize the correspondent point x' in the view $j + 1$, from the point x in the view j , and using the Optical Flow $h_-^{j+1}(x')$ in the opposite direction from the view $j + 1$ to the view j to come back from the point x' in the view $j + 1$ to a point x'' in the view j . Then we have, from the Optical Flow definitions, $x' = x + h_+^j(x)$ and $x'' = x' + h_-^{j+1}(x') = x + h_+^j(x) + h_-^{j+1}(x + h_+^j(x))$. In ideal conditions, both points, x and x'' must be the same point, but in real conditions, this is usually false, due to several reasons as limitations to calculate the Optical Flow, imprecision in the Camera calibration, numerical errors, etc. In those cases, we have a distance between these points $d(x, x'') = \|x'' - x\|$. With this idea we can construct sequences of couples of points from every couple of images. Then the distance $d(x, x'')$ can be used as a criterion of quality.

We can extend this idea to obtain sequences with more than two points. One point x_j , in the view j , has its corresponding point $x_{j+1} = x_j + h_+^j(x_j)$, in the view $j + 1$, and comes back to the point $x'_j = x_{j+1} + h_-^{j+1}(x_{j+1})$ with an error $d_j(x_j, x'_j)$ (forward and backward). To add a new point to the sequence, we use the point x_{j+1} in the view $j + 1$ and the Optical Flow $h_+^{j+1}(x)$ to go from the

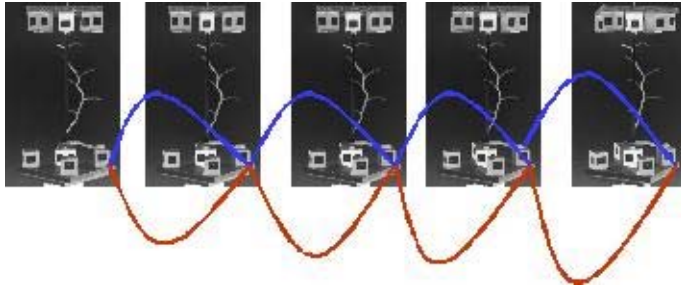


Fig. 5. Backward and Forward Optical Flow for a sequence of several points.

view $j + 1$ to the view $j + 2$. In this way we obtain, in this view, the point x_{j+2} , in correspondence with the point x_{j+1} , in the view $j + 1$. The backward Optical Flow, allows us to establish a first measure of quality of the new sequence of three points. In this way, in the view $j + 1$, we have the error forward and backward $d_{j+1}(x_{j+1}, x'_{j+1})$ where $x'_{j+1} = x_{j+2} + h_-^{j+2}(x_{j+2})$, and in the view j , the error forward and backward is $d_j(x_j, x'_j)$ with $x'_j = x'_{j+1} + h_-^{j+1}(x'_{j+1})$.

Generally, to add a new point x_{j+n} to the sequence of n points in correspondence $x_j, x_{j+1}, \dots, x_{j+n-1}$, we use the last point added to the sequence, x_{j+n-1} , and the Optical Flow $h_+^{j+n-1}(x)$ to calculate the point x_{j+n} in the view $j + n$. This point could become a new point in the sequence of corresponding points, $x_{j+n} = x_{j+n-1} + h_+^{j+n-1}(x_{j+n-1})$. To do that, we calculate the errors backward $d_i(x_i, x'_i)$ ($i = j, j + 1, \dots, j + n - 1$) where we use the Optical Flow $h_-^{i+1}(x)$ to calculate the point backward x'_i , in the view i , because $x'_i = x'_{i+1} + h_-^{i+1}(x'_{i+1})$. This point will be added to the sequence if these errors are lower than a threshold that we have defined.

We have seen that the forward and backward error is an initial quality measure of a sequence of corresponding points. The forward and backward error also allows us to establish a first test to decide whether to add a new point to the sequence or not. When we have a large number of views the initial point of a sequence will not appear on the view of the candidate point and, therefore, the forward and backward errors will be big. For a better 3D reconstruction, it is convenient that the sequence of points would be as large as possible in order to turn the algorithm less sensible to errors. We may define more complex criterions to obtain sequences with many points. Instead of using a constant threshold for the forward and backward errors, we may adapt this threshold to the sequence length. This would benefit larger sequences of points with bigger error rather than short sequences with a smaller error. Another possible criterion that would be more robust is that of using the candidate point, the sequence of corresponding points and the calibration matrices to reconstruct the 3D point and measure the reprojected errors on the projection planes. In case that the reprojected errors would be smaller than a given threshold we would include the candidate point into the sequence. It is also possible and adaptive scheme to benefit the larger sequences.

5.1 The Optimal Algorithm

The Optimal Algorithm to calculate sequences of points in correspondence is the following:

1. For every view
 - a) For every pixel
 - i. Calculate the sequence of points in correspondence and its quality
 - ii. Add the sequence of points in correspondence calculated in the previous step to an ordered repository. The first order criterion is the number of points and the second one is the quality of the sequence of points in correspondence
2. For every sequence in the sequence repository
 - a) Check that the sequence of points in correspondence is not included in another sequence of points in correspondence with a higher number of points, or with the same number of points but with a higher quality
 - i. If the sequence of points in correspondence is not included in any final repository of sequences of points in correspondence, then add the sequence to the final repository of sequences
 - ii. If the sequence of points is included, then reject the sequence

In the step 1 iterates through all views building for each pixel on the view a sequence of corresponding points and assigning a quality value. n views will produce $n * widthview * heightview$ sequences of corresponding points. In the step 2 the sequences of corresponding points are selected. Normally some of the sequences will be included on larger sequences or they would be as close to consider that they are practically the same sequence. Although this algorithm constructs large sequences of corresponding points, it has the disadvantage that it is computation costly. Step ... takes into account that a new sequence will not be introduce into the repository if the sequence already exists. At the beginning when the repository size is small it is fast to test if a sequence is already included, but when the size is increased the number of the comparisons will also increase making the process much slower.

5.2 The Fast Algorithm

To decrease the computational cost of the efficient algorithm presented above we modify the algorithm by adding a flag to every pixel. This flag indicates if a pixel is included in a sequence of corresponding points. The algorithm would be as follows:

1. For every view
 - a) For pixel on the view not included in a sequence
 - i. Compute the sequence of corresponding points. Mark every point in the sequence as included in a sequence of corresponding points
 - ii. The computed sequence in the previous step is added to a repository

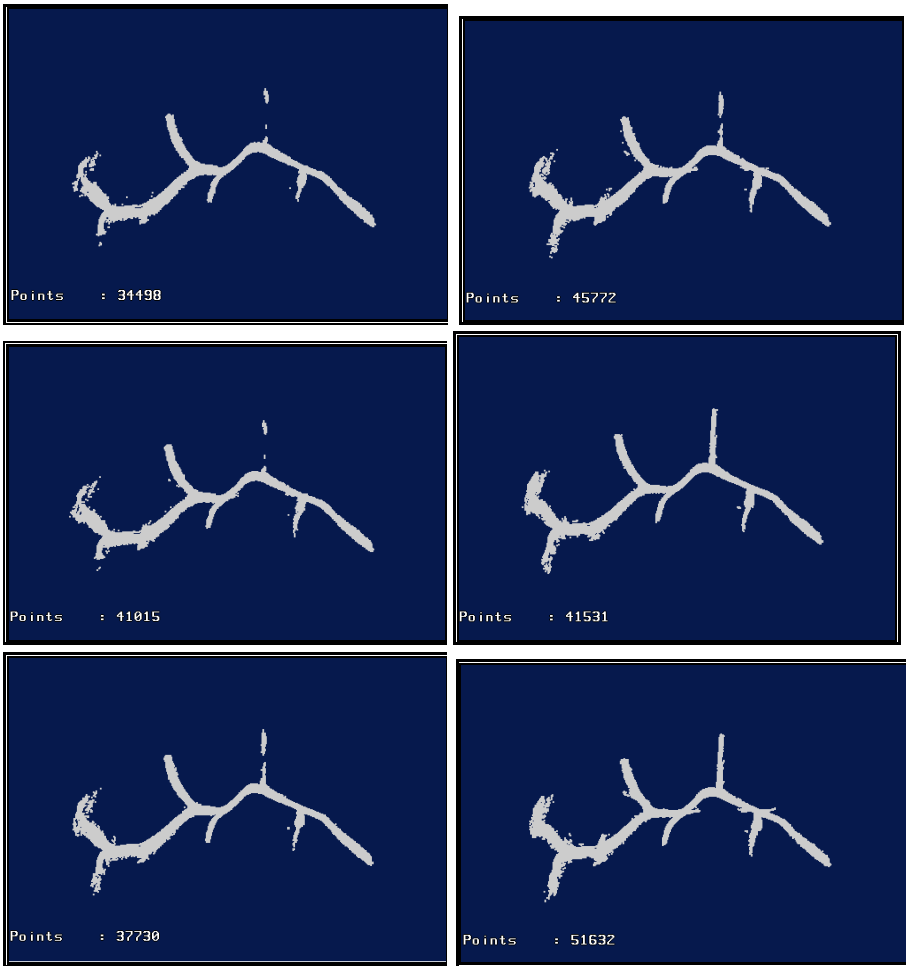


Fig. 6. 3D Reconstruction in the first line we have the results with 16 views in the second line we have the results with 25 images and from left to right we have the image with a projection error 1.0, 1.2 and 1.5.

Contrary to the efficient algorithm, the fast algorithm does not guarantee that the sequences of corresponding points would contain the larger number of points.

When we have a set of images, then we take a couple of cameras by chance and we reconstruct the 3D point. We project this point to the plane of the whole cameras and we calculate the projection error, the distance between both points, the real point and the projected point in every camera.

Later we take another couple of cameras, always by chance, and we repeat the same steps several times, up to 8 times if it is possible. We keep with the point that minimizes the projection error.

6 Results

In order to obtain the 3D reconstruction we have used two set of images, first set has 16 images, and second set has 25 images. And we have used three different threshold of projection error: 1.0, 1.2 and 1.5. In figure 6, we can see the different 3D reconstruction.

When we use more views to make the 3D Reconstruction, we have more points, but the time to do it is bigger. In the other hand when the projection error increases, the number of points increases too, but the points are located with less precision.

References

1. Luis Alvarez, Joachim Weickert, and Javier Sanchez. Reliable estimation of dense optical flow fields with large displacements. *International Journal of Computer Vision*, 39(1):41–56, 2000.
2. S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Computing Surveys*, 27(3):433–467, 1995.
3. O. Faugeras. *Three-Dimensional Computer Vision : A Geometric Viewpoint*. MIT Press, Cambridge, MA, USA, 1993.
4. Olivier Faugeras and Renaud Keriven. Complete dense stereovision using level set methods. *Lecture Notes in Computer Science*, 1406:379+, 1998.
5. R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, 2000.
6. J. Op de Beek R. Koppe E. Klotz J. Moret, R. Kemkers and M. Grass. 3d rotational angiography: Clinic value in endovascular treatment.
7. C. Cuenca L. Álvarez and L. Mazorra. Calibración de multiples cámaras utilizando objetos de calibración esféricos. In Antonio Bahamonde Rionda and Ramón P. Otero, editors, *IX Conferencia de la Asociación Española para la Inteligencia Artificial and IV Jornadas de Transferencia Tecnológica de Inteligencia Artificial, CAEPIA-TTIA 2001*, pages 1281–1290. Asociación Española para la Inteligencia Artificial and Centro de Inteligencia Artificial de la Universidad de Oviedo, 2001.
8. C. Cuenca J. Esclarín L. Álvarez, K. Baños and J. Sánchez. 3d reconstruction from a vascular tree model. In Alexis Quesada-Arencia Roberto Moreno-Díaz jr. and José Carlos Rodríguez-Rodríguez, editors, *9th International Workshop on Computer Aided Systems Theory, EUROCAST 2003, Cast and Tools for Complexity in Biological, Physical and Engineering Systems, EXTENDED ABSTRACTS*, pages 105–106. Instituto Universitario de Ciencias y Tecnologías Cibernéticas de la Universidad de Las Palmas de Gran Canaria, 2003.
9. Hans-Hellmut Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from images sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593, 1986.
10. B. D’Herde-L. Storme J. Vanrusselt P. Peene, P. Cleeren and G. Souverijns. Non substracted rotational angiography on a multipropose digital carm radiography system.

11. L. Robert and R. Deriche. Dense depth map reconstruction: A minimization and regularization approach which preserves discontinuities. In B. Buxton and R. Cipolla, editors, *Computer vision - ECCV'96, Volume I, Lecture Notes in Computer Science, Vol. 1064*, pages 439–451.
12. E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice-Hall, Inc, Upper Saddle River, NJ, 1998.
13. J. Weickert. Anisotropic diffusion in image processing.

ESKMod, a CommonKADS Knowledge Model Integrating Multiple Classic Edge Based Segmentation Algorithms

Isabel M. Flores-Parra and J. Fernando Bienvenido

Dept. Languages and Computer Science,
University of Almería,
E-04120 La Canada (Almería), Spain.
{iflores, fbienven}@ual.es

Abstract. Bibliography offers us a wide set of image segmentation methods, which are usually oriented to solve specific problems. These are evaluated, mainly, in specific work fields, as the analysis of industrial pieces, classification of agricultural products or remote sensing. Actually, main efforts are concentrated in the definition of new algorithms, generating a wider collection of alternative methods. Taking account of our experience about applying segmentation methods to diverse work fields, we realized the advantages of integrating previously used methods into a single segmentation model, with multiple alternatives. In this work, we present how some classic methods are integrated into a single segmentation knowledge model, using the CommonKADS modeling tools. Our final objective is building a model of the applicability of the alternatives in order to relate them with searched results. Here, we present how this composed knowledge model was assembled.

1 Introduction

As we found when started analyzing images in three different work areas (product classification, analysis of defective pieces and remote sensing), knowledge about image segmentation is sparse. We found in the bibliography multiple segmentation methods, but usually, they were adapted to specific problems, situations and even set of images. Main objective of automating the analysis of the images offers great difficulties, due to the fact that the interesting objects are masked by noised and negligible objects. Main problems are detecting objects from the noise and discriminating the work objects from the uninteresting ones. This later problem is specific of each area, as the interesting objects of an application field can present the characteristics of the uninteresting ones in another application field. Intrinsic complexity of the task of segmentation and discrimination of objects forces the role of human experts who handmakes or supervise these tasks. Actually, some modern methods try to substitute the action of the human expert assembling knowledge models of their action or using auto learning approaches for specific problems.

A single tool that could segment images in a general way would be an extremely useful but complex system. Our proposal, in this sense, is developing a single model of the previously defined methods that could be used as the origin of an ontology of

methods, which could be reused partially analyzing their applicability from a general point of view.

This idea of assembling a general knowledge model about the application of segmentation alternatives had its origin in a set of works related with the development of algorithms to segment same field images with objects at different scales and with different interrelations. Developing ESKMod is the later step of an evolutive process, which previous phases were:

1. Analysis of the applicability of a segmentation algorithm by Cayula and Cornillon [1] [2] [3] (from now, original algorithm) to images containing lower scale objects. The original algorithm proposed these steps:

- Preprocess the image, reducing noise and masking uninteresting details.
- Divide the image in a set of windows of equal size, covering the whole image. Authors proposed an empirical single window size.
- Filter windows:
 1. By bimodality - It works only with bimodal windows.
 2. By cohesion conditions. Equal valued pixels must be grouped. It uses three conditions over both populations (we work only with bimodal windows):

$$C_A = R_A / T_A \geq 0,92 \quad C_B = R_B / T_B \geq 0,92 \quad C = (R_A + R_B) / (T_A + T_B) \geq 0,90 \quad (1)$$

- Fix borders inside the filtered bimodal windows.
- Complete image using contour algorithms.

This algorithm, specifically oriented to the detection of the north border of the Gulf Stream, was tested in order to detect medium size structures, offering inadequate results.

2. Weakening of the restrictions of the original method. In order to increase the number of segments detected, original algorithm was modified in these points:

- Elimination of the bimodality condition. It works with multiple population windows.
- Use of three sets of windows of different sizes (in order to detect different size objects).
- Filter windows by the cohesion conditions dividing the pixels in two populations using a mobile limit.
- Fix border for the first two population distribution of pixels trespassing limits.

This version improved the detection of medium size objects, but it did not detect little size objects, generating sometimes borders with different threshold by section.

3. Threshold analysis and initial parameterization. In order to fix the restrictions and problems of previous algorithms, we proposed [4] a generalization of the previous methods, offering some threshold options and analyzing their impact in the results. Main changes were:

- It was explicitly formulated the one border maximum per window criterion.
- It was proposed to analyze the threshold dividing the pixels of each window in two populations. All possible thresholds would be studied, generating a set of possible pixel distributions (and subsequently different alternative borders), as shown in fig. 1.

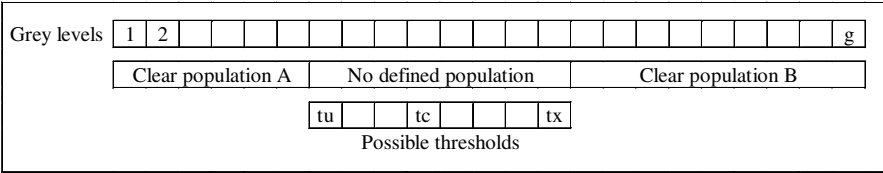


Fig. 1. Analysis of possible thresholds over a generalization of the Cayula algorithm

- Three possible threshold selection options were defined:
 - LL- Select the lowest limit (*tu*). Contours will limit dark zones.
 - UL- Select the uppermost limit (*tx*). Contours will limit light zones.
 - OL- Select the limit (*tc*) where global cohesion (*C*) is maximal. Contours will limit different structures, locating borders with maximal differences.

This version offered better results for images with little size objects. Main problem was how to control of different size objects and the restriction of the parameterization to only one variable.

4. Problem parameterization and building of multiple algorithmic options. We developed multiple alternatives of application of the algorithm [5], analyzing the impact of modifying these parameters:
 - Pretreatment filters.
 - Window sizes.
 - Combinations of set of windows sizes.
 - Overlapping of windows of each set size.
 - Compacity limits.
 - Threshold definition (as they were defined in previous step).
 - Application of morphological treatment over the generated bimodal windows previously to the border generation.

We generated a wide set of alternatives, in order to segment images with different combinations of objects. Figure 2 shows the tree of options for the pretreatment 1. We generated 72 alternatives (using a fixed set of windows sizes), which offered different results.

5. Quantitative evaluation of the different alternatives. Next work was ordering all the generated results evaluating their usefulness [6] [7]. We defined [5] a set of metrics to compare different segmented images between them (it was a relative evaluation). This set of metrics was assembled using these principles:
 - A **metric** is defined as an objective (as opposed to subjective) value computed using the edges located on a given image and the characteristics of its computing process. We defined a wide set of metrics, classified using two dimensions: scope and characteristics as shown in table 1.
 - We evaluated the results for specific working areas, pixel values and type of objects analyzed. As an example, a metric that count the number of border pixels for a given area (*s1*≡ the whole image), a range of values (*f2*≡ temperatures), and some sort of objects (*o1*≡ all the objects), will be referenced as *Vs1f2o1*. It is possible to use metrics of type *Xs-f-o** when it is available some information about the family *** of objects.

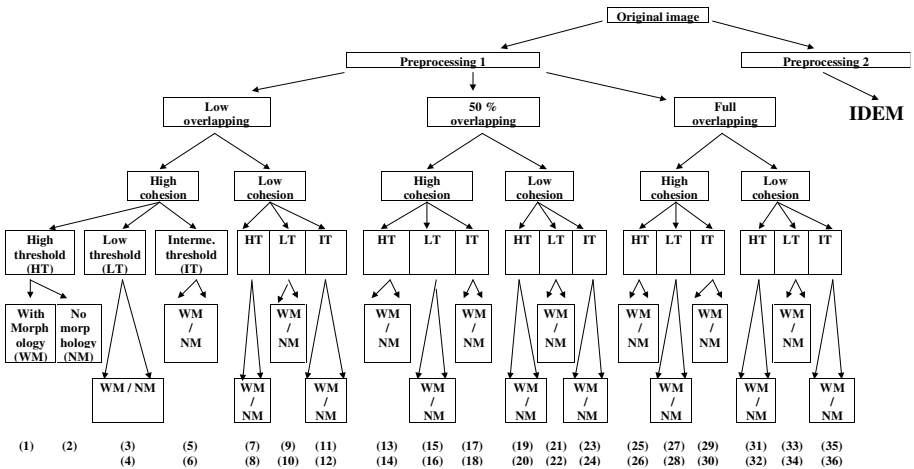


Fig. 2. Tree of algorithmic alternatives generated for a single pretreatment option

Table 1. Basic set of metrics

<i>Characteristics</i>	<i>Scope</i>	Space (s1, s2,...)	Range (f1, f2,...)	Object (o1, o2,...)
Volume		Vs	Vr	Vo
Quality (length)		Ls	Lr	Lo
Quality (width)		Ws	Wr	Wo
Quality (continuity)		Cs	Cr	Co
Specificity		Ss	Sr	So
Regularity		Rs	Rr	Ro
Use of resources		Us	Ur	Uo

- All the different metrics have been defined analytically, as shown for metric Vs1f2o*:

$$Vs1f2o^* = \forall (x,y) \in s1 \sum (pixborder(x,y) * valmask(x,y)) \quad (2)$$

$$\begin{aligned} pixborder(x,y) &= \begin{cases} 1 & \text{If } (x,y) \text{ is a border pixel} \\ 0 & \text{If } (x,y) \text{ is not a border pixel} \end{cases} \\ valmask(x,y) &= \begin{cases} 1 & \exists (x_\alpha, y_\alpha) \in E(x,y) / \text{imagen}(x_\alpha, y_\alpha) \in f2 \\ 0 & \forall (x_\alpha, y_\alpha) \in E(x,y) / \text{imagen}(x_\alpha, y_\alpha) \notin f2 \end{cases} \end{aligned}$$

$$E(x,y) = \{(x_\beta, y_\beta) \in s1 / x+1 \geq x_\beta \geq x-1, y+1 \geq y_\beta \geq y-1\}$$

- Using a combination of this metrics in a multicriteria function (MF), it is possible to classify different results. This way, we have a single value that let us to select automatically best results.

Using these metrics, we could not conclude which version and parameterization of the algorithm offered best results; different images required different versions. Metrics helped to evaluated options for specific cases, but not gave order to the increasing set of alternatives.

From all these precedent works, we concluded the requirement of a model of applicability of the different versions and possible parameterizations of the algorithm. We decided to assemble a knowledge model about how segmentation is done, integrating all the previous options and using the metrics as future selection criteria. Starting from the set of methods previously tested, we realized the advantages of extending the work space to the different segmentation and image analyzing methods (in the sense of preparing its place into the model). Showing how this model is being assembled is the objective of this work.

2 Methodology

In order to build the knowledge model about the image analysis and segmentation we have to follow some steps: first, selecting a knowledge modeling methodology evaluating its appropriateness; second, fixing the scope of the model ordering the methods to be included in an initial phase and later; and finally, assembling a common framework to integrate different and disperse methods in a general schema.

2.1 Knowledge Modeling Methodology

Our main objective was to develop a knowledge based model of the image analysis and segmentation processes, as a de facto standard in Europe; we decide to use the knowledge modeling tools of CommonKADS [8] [9]. Our initial objective was developing the knowledge model, not to implement a knowledge based system, but this could be a next step, once the knowledge based model would be assembled, as it is actually. In order to build the knowledge model, we have used, as modeling tools mainly, the task-method diagrams and CML descriptions. Original notation elements were modified [10] in order to simplify diagrams and descriptions, facilitating the construction of an ontology (including the different knowledge elements). Main changes were:

- Fixing a set of restrictions when assembling task-method diagrams (TMD):
 - Every task can be solved -formally- by a set of alternative methods, even in the case we have found only one by the moment. We prepare always a set of processing alternatives for each objective.
 - Methods can be simple or composed.
 - Each simple method is related with a single primitive; the method supposes the application of this primitive.
 - Composed methods are partitioned in partial objectives, represented by subtasks or transfer functions (when their objectives are reached, respectively, into or outside the knowledge agent).
- Defining a new type of diagrams, extended task-method diagrams (ETMD), that integrates into the previously defined TMD some control elements (based in the Jackson diagrams notation elements [11]). New elements of the ETMD included in this extension are shown in figure 3, and new relations in figure 4.



Fig. 3. ETMD new notation elements

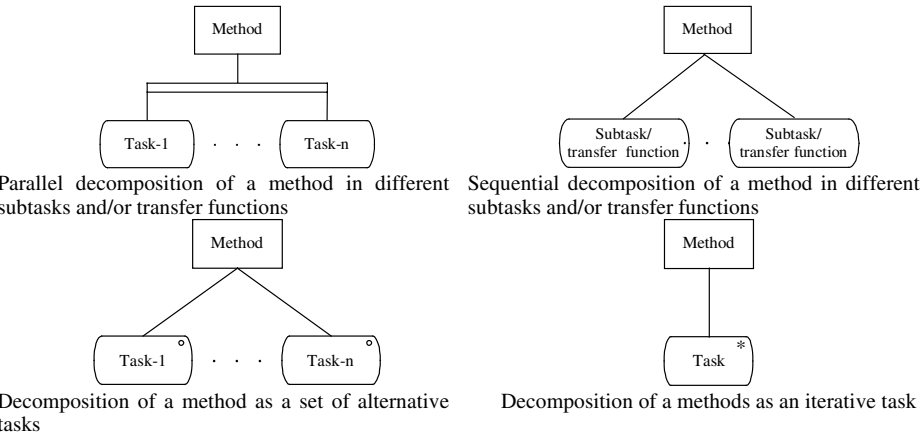


Fig. 4. ETMD new notation elements

One key point of the model to be assembled was the existence of multiple alternatives. We proposed to manage them using the mechanism of the dynamic selection of methods developed originally by Benjamins [12] and reformulated later by us [13] [14] [15] with the use of these components:

- *Suitability criteria*. They are specifically associated to each method, modelling the different characteristics of interest when selecting the method.
- *Suitability criteria weights*. Specific of each task, they show the relative importance of each suitability criteria for the given task.
- *Selection data and knowledge*. Domain knowledge elements used in the selection process.
- *Aggregation function*. It integrates the different suitability criteria and weights, generating a single suitability value for each assigned method.
- *Selector*. This general computing element manages all other elements for a given task. This selector is general.

Summarizing, we modelled the task of image segmentation by border detection using an extension of the modelling tools of CommonKADS plus the elements of the dynamic selection of methods in order to manage the different alternatives.

2.2 Model Scope

Our main objective was modeling the task of segmentation of images by border detection as part of a more general image analysis task. First, we decided to analyze

its context. We segment images in order to extract from them some sets of objects (areas or structures) that are useful in our specific application. In order to understand the restrictions and of the modeled task and prepare the extension of the model to other related task, we modeled the general image analysis task as shown in figure 5.

From a general abstracted point of view, analyzing the image supposes to extract from the original digital image a set of objects (areas or structures) that are filtered (maintaining only those of interest) and somehow classified (as it is the case of the detection of defects in industrial pieces or burned areas in a satellite image).

Segmenting the image supposes to divide the image totally or partially (it is possible to obviate directly some elements, areas or structures) in independent sections. There are two main mechanisms to do this:

- Fixing the borders between these sections.
- Giving specific values to the pixels composing these regions.

Users can be interested in different kinds of objects, areas or structures (delimited by borders with specific shapes), requiring to extract this features from the segmented images. All the objects extracted are not usually of interests; only some of them are useful in multiple cases (as it is the case of critical defects in industrial pieces or ready to pick tomatoes for an automatic collecting system). It is required a task to filter the set of generated objects.

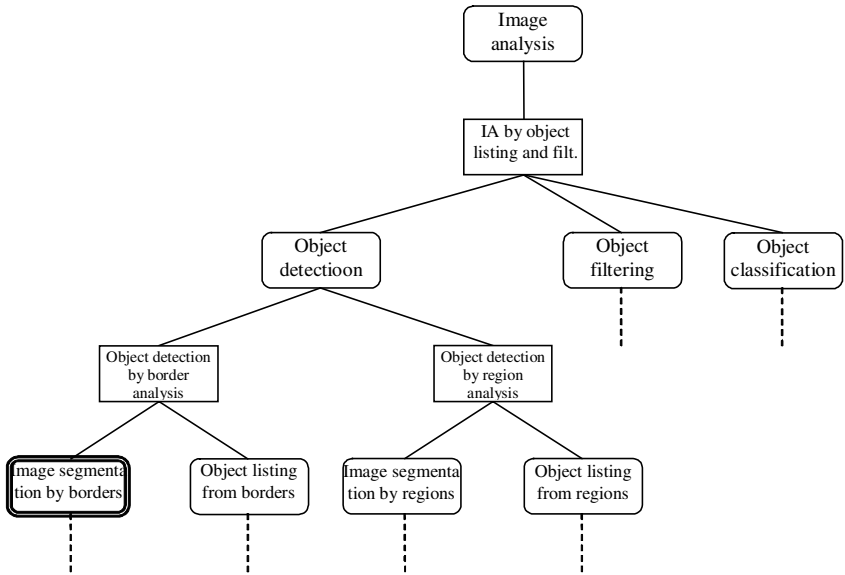


Fig. 5. Context of the task of image segmentation by border detections

Finally, filtered objects are usually classified in order to execute later actions (as deviating to alternative boxes different quality agronomical products from a classification chain).

In this work, we analyze a model restricted to the image segmenting task by border detection, integrating part of the actual segmentation methods found in bibliography, rewritten and tested by ourselves. So, we restricted first our work domain to the edge

oriented methods [1] [2] [3] [16] [17] [18] [19] [20] [21] [22], because of the requirement of outlining the problem and our previous experience applying this sort of methods. Next, we selected a set of classic segmentation methods to be analyzed and integrated into knowledge mode about the segmentation of image to be developed. We selected some classic segmentation algorithms as marked in figure 6.

Classic segmentation methods	Border detection	• Gradient operators • Mathematical morphology • Mobile windows • Coocurrence matrixes	<i>Modeled</i> <i>Modeled</i>
	Region detection	• Texture analysis • Seed methods • Region merging • Region splitting	 <i>Variation</i> <i>(for edges)</i>

Fig. 6. Classification of classic segmentation methods pointing those included in ESKMOD

We integrated mainly, taking account of our previous experience, some methods using mobile windows, mathematical morphology, and a variation of the seed methods modified by ourselves. Later works will include other groups of methods and other alternatives of same groups.

2.3 Integration of Methods

Next, our work focused in organizing multiple previous segmentation algorithms in order to develop a single models, where selecting alternative options we can obtain the behavior of the different original algorithms. Specifically, we divide each different algorithm in elementary parts, where each one transforms anyway the previous image or data about it. In this sense, we defined, as shown in figures 7 to 8 a set of 22 general steps, which can be activated optionally into the different algorithms.

Taking account of all these different possible actions, we analyzed all the work methods in order to fix the possible activation of an algorithmic alternative of each step. Each studied alternative algorithm supposes the activation of a subset of these actions using specific methods in order to solve them. Integrating these steps, we defined a composed method that summarized the behavior of all previous ones.

3 Results

Main result is a knowledge based model of the task of segmentation by edge detection, called ESKMod, integrating different approaches located in the bibliography and sometimes modified by ourselves. Figures 9 and 10 show the form of some ETM diagrams and the CML description of an element of the model.

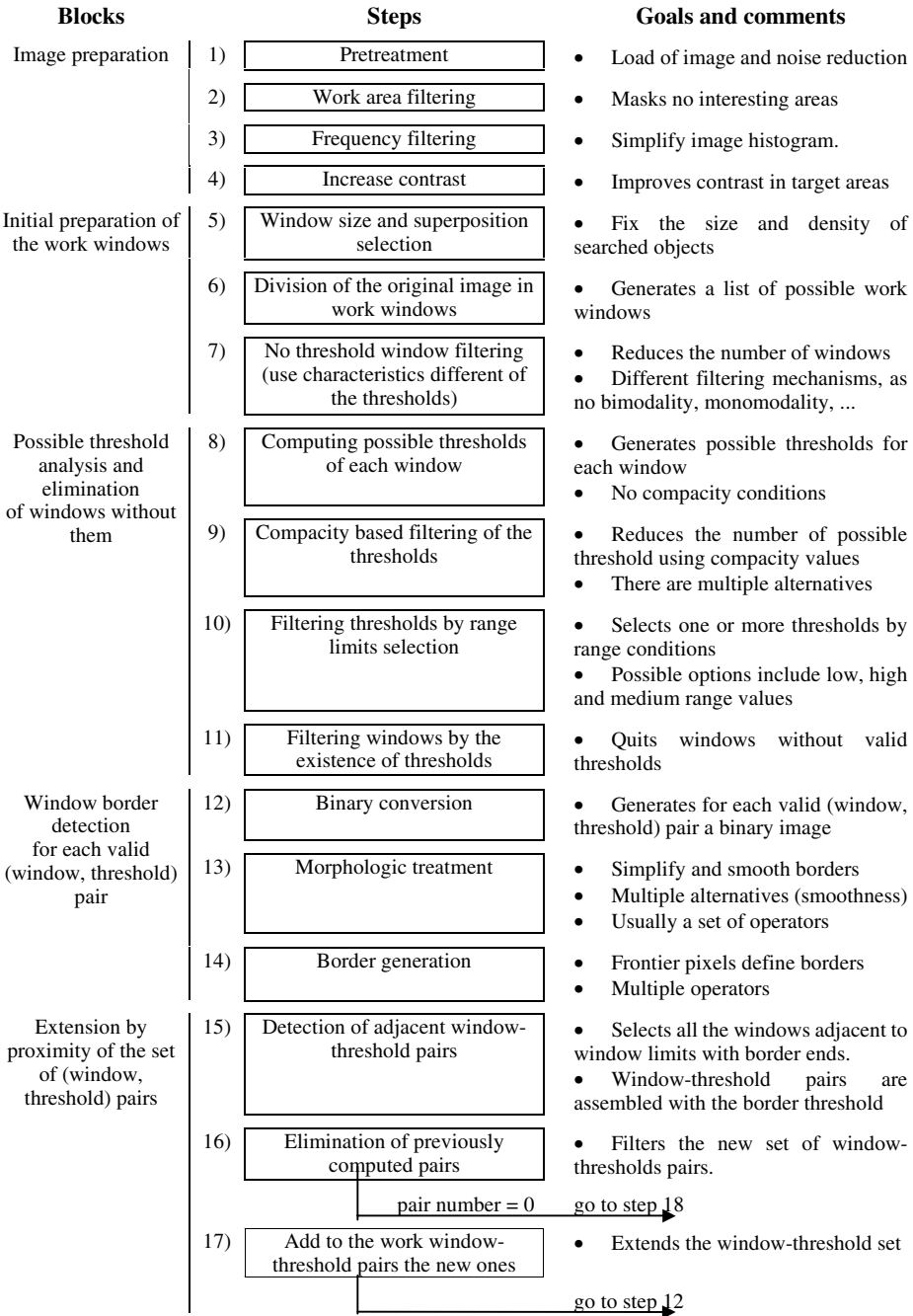


Fig. 7. Elementary steps used optionally by analyzed algorithms (1)

Blocks	Steps	Goals and comments
Composition of resulting borders	18) Combination of the resulting borders by window size	Generates a global image for each window size
	19) Combination of global images for the different window sizes	Obtains a single image with all the generated
Border improving	20) Completion of border segments	- Connects separate border segments - Less useful when extended the set of pairs window-threshold
	21) Elimination of spurious borders	Eliminates little unconnected edges
Results presentation	22) Combination of the border and original images	- Superposes images only at the presentation level - It would be possible to analyze only border or superpose images

Fig. 8. Elementary steps used optionally by analyzed algorithms (2)

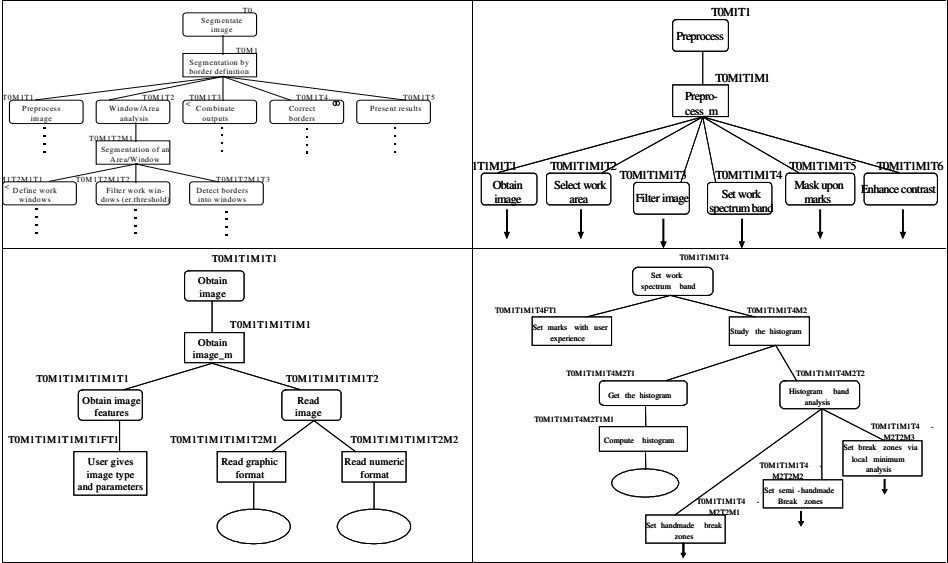


Fig. 9. ETM diagrams of ESKMod

4 Conclusions and Future Works

Developing the knowledge model about segmenting images by edge detection let us to order multiple ideas extracted from the bibliography, being the base of a new software system (actually under development).

TASK *Segment Image by edge detection*
 GOAL: "Segment raster image";
 ROLES:
 INPUT: in-image: "original raster image";
 OUTPUT: out-image: "final raster image, it contains borders or specific regions related with the specific problem";
 SELECTION-CRITERIA:
 END detect_b = "Use a border detection algorithm vs a region detection one";
 CRITERION-WEIGHTS:
 AGREGATION-METHOD: SM;
 SPECIFICATION: "Given a original raster image, it obtain a new image where there are present region borders or specific regions";
 ASSUMPTIONS: "Raster image corrected in order to be used in the segmentation process, e.g. georeferenced";
 END TASK *Segment Image*

Fig. 10. Example of CML description. Task *segment image*

Main conclusions of the actual work are:

1. Modeling the segmentation of images by edge detection (as part of the image analysis process) has been useful ordering multiple ideas oriented usually to specific cases and distributed by the bibliography.
2. The integrated model let us to reuse multiple elements of the different alternative algorithms, stored into an ontology, as a base of a powerful composed method.
3. ESKMod includes a model of applicability, integrating data from bibliography and metrics of multiple tests.
4. CommonKADS tools adequately extended has proved as a powerful framework in order to develop ESKMod.

Actual and future developments include:

1. Implementing practically the composed method integrating all the elements included into ESKMod (actually under development).
2. Extending the scope of the model to other methods as those that segment images by region.

References

1. Cayula, J.F.: Edge detection for SST images. M.S. Thesis, Dept. of Electrical Engineering, Rhode Island Univ. (1988)
2. Cayula, J.F., Cornillon, P.: Comparative Study of Two Recent Edge-Detection Algorithms Designed to Process Sea-Surface Temperature Fields. IEEE Transactions on Geoscience and Remote Sensing, Vol. 29, N. 1 (1991)
3. Cayula, J.F., Cornillon, P.: Edge detection algorithm for SST images. Journal of Atmospheric and Oceanic Technology (1992)
4. Flores-Parra, I.M., Bienvenido, J.F.: Selection of thresholds on an edge detection algorithm based on spatial compacity. Proc. IGARSS'99 (1999) 125–127.
5. Flores-Parra, I.M., Bienvenido, J.F., Menenti, M.: Characterization of segmentation methods by multidimensional metrics: application on structures analysis. Proc. IGARSS'00, IEEE 0-7803-6362-0/00, Hawaii (2000) 645–647
6. Phillips, I.T., Chhabra, A.K.: Empirical performance evaluation of graphics recognition systems. IEEE Trans. Pattern Anal. Mach. Intell., Vol. 21, N. 9 (1999) 849–870

7. Mao, F., Gill, J., Fenster, A.: Technique for evaluation of semi-automatic segmentation methods. *Proc. SPIE, Int. Soc. Opt. Eng.*, Vol 3661 (1999) 1027–1036
8. Schreiber, G. et al., “CommonKADS: A comprehensive methodology for KBS development”, *IEEE Expert*, 9–6 (1994) 28–37
9. Schreiber, G., Akkermans, J.M., Anjewierden, A.A., de Hoog, A., van de Velde, W., Wielinga, B.J.: *Knowledge Engineering and Management. The CommonKADS Methodology*. MIT Press (1999)
10. Iglesias, C.A., Garijo, M., González, J.C., Velasco, J.R.: A methodological proposal for multiagent systems development extending CommonKADS. In: Gaines and Musen (eds.): *Proc. of KAW'96*, Vol. I (1996) 25–1/17
11. Jackson, M.: *System Development*. Prentice-Hall (1983)
12. Benjamins, V.R.: *Problem Solving methods for Diagnosis*. Doctoral Thesis, University of Amsterdam (1993)
13. Bienvenido, J.F.; Flores-Parra, I.M.; Marin, R.: Dynamic selection of methods based on the aggregation of suitability criteria and weights. *Proceeding of the IFIP World Computer Congress 2000, Beijing–China Vol. K (2000)* 407–412
14. Bienvenido, J.F., Flores-Parra, I.M., Guirado, R., Marín, R.L.: Knowledge based modeling of the design process as a base of design tools. *Computer Aided System Theory – EUROCAST 2001, Lecture Notes in Computer Science*, Vol. 2178 (2001) 207–222
15. Bienvenido, J.F., Flores-Parra, I.M.: Extended models of dynamic selection using ontological elements. Application to design and image analysis problems. *KES 2003. Lecture Notes in Computer Science (in press)*
16. Heath, M., Sarkar, S., Sanocki, T., Bowyer, K.: Comparison of Edge Detectors. *A Methodology and Initial Study. Computer Vision and Image Understanding*, Vol. 69, N. 1 (1998) 38–54
17. Simpson, J.J.: On the Accurate Detection and Enhancement of Oceanic Features Observed in Satellite Data. *Rem. Sens. Env.*, N. 33 (1990) 17–33
18. Szczechowski, C.: The Marr-Hildreth Operator as an Eddy Detector. *Proceedings: Automated Interpretation of Oceanographic Satellite Images Workshop*, SP 001:32191 (1991) 153–168
19. Holyer, R., Peckinpugh, S.: Edge detection applied to satellite imagery of the oceans. *Trans. on Geoscience and Remote Sensing*, Vol. 27, N. 1 (1989) 46–56
20. Lea, S.M., Lybanon, M.: Automated Boundary Delineation in Infrared Ocean Images. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 31, N. 6 (1993) 1256–1260
21. Krishnamurthy, S., Iyengar, S.S., Holyer, R., Lybanon, M.: Histogram-based morphological edge detector. *IEEE Trans. on Geoscience and Remote Sensing*, Vol. 32, N. 4 (1994) 759–767
22. Flores-Parra, I.M., Bienvenido, J.F., Menenti, M.: Detection of Structures using a Robust Edge Detectin Algorithm with Expansion of Border Windows. *Proc. IGARSS'02 (2002)*

Frequency Analysis of Contour Orientation Functions for Shape Representation and Motion Analysis

Miguel Alemán-Flores, Luis Álvarez-León, and Roberto Moreno-Díaz jr.

Departamento de Informática y Sistemas
Universidad de Las Palmas de Gran Canaria
35017 Las Palmas, Spain
{maleman,lalvarez,rmorenoj}@dis.ulpgc.es

Abstract. The location and characterization of edges is a crucial factor for some image processing tasks, such as shape representation and motion analysis. In this paper, we present a set of filters which allow estimating edge orientation and whose output does not vary when the input signal is rotated or when a global illumination change occurs. The outputs of these filters are used to extract a one-dimensional representation of the contour in order to characterize shapes in an accurate way by using Fourier coefficients. An energy function is built to associate shapes and determine the similarities between object contours. The elements of this energy function are weighted so that those which provide the most relevant information have a larger contribution. By extracting the parameters of the transformations which relate every image in a sequence with the following, we can determine the temporal evolution of a moving object.

1 Introduction

In this work, we present a model for analyzing shape and motion by using a set of filters and primitives to estimate edge orientation. We introduce a set of formal tools, based on Newton filters [1][2], which allow estimating edge orientation and whose output does not vary when the input signal is rotated or when a global illumination change occurs. The outputs of these filters allow us to extract a one-dimensional representation of the contour in order to characterize shapes by using Fourier coefficients.

The Fourier analysis of the orientation functions allows extracting some general features of the shape which is represented. Moreover, we propose to use weighting functions for the coefficients of the different frequencies in such a way that those coefficients whose information is more relevant have a higher weight in the resulting scheme. We analyze the shapes of such weighting functions in order to improve the discrimination and reduce the failure probability. We have applied these new techniques to a database consisting of 1000 shapes of marine

animals and we have obtained a very satisfactory classification in a quite difficult situation. We propose to use the information extracted for a segment of an orientation function to relate different parts of a signal in such a way that we can associate the visible regions of an object when it is not completely accessible.

The numerical experiences are very promising. In particular, we can even discriminate between shapes which are very similar from a perceptual point of view. Furthermore, the accuracy of edge orientation estimations have made it possible to enlarge the range of applications to those in which the refinement of these values favors considerably the parameter extraction, such as motion analysis.

This work is structured as follows: In section 2, Newton filters are revisited and we show how these filters can be adapted to build a new type of filters, the modified Newton filters, which can be used for the estimation of edge orientation. In section 3, we describe how we can build a representation of the contour of an object from the outputs of these new filters and how this representation can be analyzed from its Fourier coefficients. We introduce the energy function which is used to measure the similarity between two shapes and decide whether two objects belong to the same shape category or not. In section 4, the information extracted from the contours is used to fit objects and analyze their motion, obtaining translation, rotation and scaling parameters. Finally, in section 5, a brief discussion about the previous topics is presented, including the most relevant conclusions extracted from this work.

2 Orientation Selective Edge Detectors

In this section, we use a new set of filters which preserve the convenient properties of original Newton filters, but which also avoid some of the undesirable phenomena. They are shown in table 1. In particular, filter M_k reacts to changes in the orientation $k\frac{\pi}{4}$.

While in the original Newton filters we observed some differences according to the orientation of the edge we wanted to detect, in this set, the weights are arranged cyclically. This implies that rotating the image a multiple of $\pi/4$ does not alter the magnitude of the output, but only the indices.

Table 1. Modified Newton filters and corresponding orientation

<table><tr><td>1</td><td>1</td><td>-2</td></tr><tr><td>2</td><td>2</td><td>-4</td></tr><tr><td>1</td><td>1</td><td>-2</td></tr></table>	1	1	-2	2	2	-4	1	1	-2	<table><tr><td>1</td><td>-2</td><td>-4</td></tr><tr><td>1</td><td>2</td><td>-2</td></tr><tr><td>2</td><td>1</td><td>1</td></tr></table>	1	-2	-4	1	2	-2	2	1	1	<table><tr><td>-2</td><td>-4</td><td>-2</td></tr><tr><td>1</td><td>2</td><td>1</td></tr><tr><td>1</td><td>2</td><td>1</td></tr></table>	-2	-4	-2	1	2	1	1	2	1	<table><tr><td>-4</td><td>-2</td><td>1</td></tr><tr><td>-2</td><td>2</td><td>1</td></tr><tr><td>1</td><td>1</td><td>2</td></tr></table>	-4	-2	1	-2	2	1	1	1	2
1	1	-2																																					
2	2	-4																																					
1	1	-2																																					
1	-2	-4																																					
1	2	-2																																					
2	1	1																																					
-2	-4	-2																																					
1	2	1																																					
1	2	1																																					
-4	-2	1																																					
-2	2	1																																					
1	1	2																																					
$M_0 : 0$	$M_1 : \pi/4$	$M_2 : \pi/2$	$M_3 : 3\pi/4$																																				
<table><tr><td>-2</td><td>1</td><td>1</td></tr><tr><td>-4</td><td>2</td><td>2</td></tr><tr><td>-2</td><td>1</td><td>1</td></tr></table>	-2	1	1	-4	2	2	-2	1	1	<table><tr><td>1</td><td>1</td><td>2</td></tr><tr><td>-2</td><td>2</td><td>1</td></tr><tr><td>-4</td><td>-2</td><td>1</td></tr></table>	1	1	2	-2	2	1	-4	-2	1	<table><tr><td>1</td><td>2</td><td>1</td></tr><tr><td>1</td><td>2</td><td>1</td></tr><tr><td>-2</td><td>-4</td><td>-2</td></tr></table>	1	2	1	1	2	1	-2	-4	-2	<table><tr><td>2</td><td>1</td><td>1</td></tr><tr><td>1</td><td>2</td><td>-2</td></tr><tr><td>1</td><td>-2</td><td>-4</td></tr></table>	2	1	1	1	2	-2	1	-2	-4
-2	1	1																																					
-4	2	2																																					
-2	1	1																																					
1	1	2																																					
-2	2	1																																					
-4	-2	1																																					
1	2	1																																					
1	2	1																																					
-2	-4	-2																																					
2	1	1																																					
1	2	-2																																					
1	-2	-4																																					
$M_4 : \pi$	$M_5 : 5\pi/4$	$M_6 : 3\pi/2$	$M_7 : 7\pi/4$																																				

When the real orientation does not correspond to one of the 8 main directions, we can estimate it by interpolating the higher value in the output vector F with its two neighbors, which provides a much more accurate estimation of the orientation. The main advantage of this new kind of filters is not the location of edges, but their classification according to their orientation and the invariance under rotations and global illumination changes.

3 Shape Representation

This section deals with the problem of identifying an object from its contour. The outline is extracted by means of the filters previously described and an orientation function is built to characterize the shape. We use an energy function to measure the similarity between two objects according to the shapes of their orientation functions.

3.1 Contour-Based Shape Representation

We have described a technique to extract local information about the orientation of the contour of an object. Once we have located the edges and the orientation in every point of the contour, we can build a one-dimensional sequence which describes how the latter changes as we move along the border. If the edges are not clear enough, double borders may appear and must be eliminated before continuing.

We must first select a point of the outline and search for non-visited neighbors until we close the figure to find a representation of its orientation function. Different types of one-dimensional shape representations are described by Loncaric [3].

In order to provide these functions with a certain continuity, we forced some conditions, such as selecting values of the orientation which do not differ in more than π radians from its neighbors, except for the first and the last points of a closed curve, which are actually neighbors.

Fourier coefficients provide a discrimination function to determine how similar they are. In the pioneering work by Zahn and Roskies [4], the optimum criteria as well as the consequences of certain transformations on Fourier coefficients are studied in the case of polygonal shapes using continuous Fourier series. In this work, we generalize the results presented by Zahn and Roskies [4] to the case of uniformly distributed sample sets of points of the boundary and we use the Fourier Transform for the characterization. More recently, Rui et al. [5] describe a set of modified Fourier descriptors for shape representation. The authors use a uniformly sampled set of points of the boundary and a similarity distance based on a combination of the magnitude and phase of the Fourier coefficients.

Let f_n be a one-dimensional discrete signal corresponding to the orientation function of an object and consisting of L values, its discrete Fourier transform coefficients can be obtained as $\hat{f}_k = \frac{1}{L} \sum_{n=0}^{L-1} f_n e^{-i \frac{2\pi kn}{L}} \quad \forall k = 0, 1, 2, \dots, L-1$.

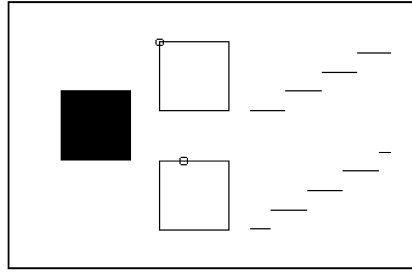


Fig. 1. Two different orientation functions for a square selecting two different starting points

To be able to compare two shapes without taking into account their sizes, the sequences obtained for orientation functions are normalized in their length, in such a way that all of them are equally long. When the object is rotated a certain angle θ , all points on the contour will undergo an increase in the values of their respective orientations. However, this increase will be the same for all of them, on the condition that it is a solid whose shape is not altered by the rotation. This phenomenon only affects order 0 coefficient of the orientation function and not any other. Let g_n be the orientation signature of the object described by f_n after a rotation of a certain angle θ , their coefficients are the same, except for order 0 coefficients.

3.2 Influence of the Starting Point of the Orientation Sequence

To extract the sequence corresponding to a closed curve, any point on the contour can be used as start. However, we must take into account the consequences that a change in the starting point produces. Firstly, a shift is observed in the values which constitute the signal. Secondly, the exigency of continuity causes an increase in the values which appear on the opposite side of the sequence. These two effects are shown in Fig. 1.

Let g_n be the signal resulting of shifting f_n a positions and increasing the values which appear on the opposite side in 2π , i.e. they correspond to the same shape starting at a different point of the contour, their coefficients are related as in equation (1).

$$g_n = f_n^a = \begin{cases} f_{n+a} & \text{if } n = 0, \dots, L-a-1 \\ 2\pi + f_{n-(L-a)} & \text{if } n = L-a, \dots, L-1 \end{cases} \quad (1)$$

This signal will generate the following coefficients:

$$\hat{g}_k = \hat{f}_n^a = \frac{1}{L} \sum_{n=0}^{L-1} f_n^a e^{-i \frac{2\pi k n}{L}} = e^{i \frac{2\pi k a}{L}} \hat{f}_k + \frac{2\pi \left(e^{i \frac{2\pi k a}{L}} - 1 \right)}{L \left(1 - e^{-i \frac{2\pi k}{L}} \right)} \quad \forall k \neq 0 \quad (2)$$

Taking into account that order 0 coefficient is related to the mean value of the signal and is altered by a rotation of the figure, it is not suitable for our purposes. On the other hand, the larger the order of the coefficient, the more sensitive it is to noise. If we only consider the relationship between order k coefficients of both signals ($k \neq 0$), we can estimate the shift as in equation (3).

$$a = -\frac{iL}{2\pi k} \ln \left(\frac{2\pi + L\hat{g}_k \left(1 - e^{-i\frac{2\pi k}{L}}\right)}{2\pi + L\hat{f}_k \left(1 - e^{-i\frac{2\pi k}{L}}\right)} \right). \quad (3)$$

Since, in most cases, the fit of the reference sequence with the extracted one will not be perfect, the value obtained for a will provide us with a first approximation.

3.3 Sequence Direction and Symmetrical Shapes Association

We have considered the problem of starting the orientation function of a closed curve at a different point, but next, we must see what happens if we choose the opposite direction to continue. This phenomenon can be detected because the difference between the first and the last point of the sequence will be positive for one of the signals and negative for the other. For closed curves, it is -2π if we progress clockwise and 2π if we do it counterclockwise.

If we are working with plane objects, e.g. we are trying to identify keys, they can be presented in two different forms, corresponding to both sides of the object. However, one of them is a reflected version of the other and their orientation functions can be coupled if we consider the changes they will undergo. If g_n represents a shape that is a reflected version of that represented by f_n , starting at the same point, they and their coefficients can be related as shown in equation (4), where C is a value which depends on the starting point of the contour and the symmetry axis that has been used for reflection, but which remains constant for all points inside the sequence.

$$g_n = \begin{cases} C - f_0 & \text{if } n = 0 \\ C - f_{L-n} + 2\pi & \text{if } n = 1, \dots, L-1 \end{cases} \quad \hat{g}_k = \begin{cases} -\hat{f}_0 + \frac{2\pi(L-1)}{L} + C & \text{if } k = 0 \\ -\hat{f}_k^* - \frac{2\pi}{L} & \text{if } k \neq 0 \end{cases} \quad (4)$$

Equations (2) and (4) will allow us to build energy functions for shape characterization, as described in the following section.

3.4 Shape Characterization

Once we have studied how the functions we use for the description of a contour are affected by certain transformations, we consider now the comparison between two or more such functions. When a shape has r -fold rotational symmetry and fits itself under a rotation of $2\pi/r$, those coefficients whose order is not multiple of r are null, thus avoiding to extract a right shift from them. In case we use

several coefficients to estimate the shift, instead of extracting it from only one of them, we can build an energy function as a sum of the errors for every coefficient. This will provide an accurate value for the shift as well as a similarity measure to compare shapes. From equation (2), we can determine how good the relationship is for a certain value of a and a given coefficient order k , as shown in equation (5):

$$E_k(a) = e^{i\frac{2\pi ka}{L}} \left(\hat{f}_k + \frac{2\pi}{L(1 - e^{-i\frac{2\pi k}{L}})} \right) - \left(\hat{g}_k + \frac{2\pi}{L(1 - e^{-i\frac{2\pi k}{L}})} \right) \quad (5)$$

$$= e^{i\frac{2\pi ka}{L}} \tilde{f}_k - \tilde{g}_k \quad \text{where} \quad \tilde{f}_k = \hat{f}_k + \frac{2\pi}{L(1 - e^{-i\frac{2\pi k}{L}})}.$$

If we add the terms corresponding to every coefficient with non-null index, multiplying each one of them by its respective conjugate, we obtain the following function, where $\tilde{f}_k$ and $\tilde{g}_k$ are obtained from Fourier coefficients as shown above:

$$E(a) = \sum_{k=1}^{\frac{L}{2}} \left(|\tilde{f}_k|^2 + |\tilde{g}_k|^2 - e^{i\frac{2\pi ka}{L}} \tilde{f}_k \tilde{g}_k^* - \left(e^{i\frac{2\pi ka}{L}} \tilde{f}_k \tilde{g}_k^* \right)^* \right). \quad (6)$$

And now, we should extract the value for a which minimizes this function. If we have obtained an approximation for a from one of the coefficients, we could use this value for an iterative scheme, such as Levenberg-Marquardt scheme, to extract a more accurate one.

Once we have found a measure of the similarity of two contours, a threshold must be set to decide whether they come from the same shape or not and, in case they do not, to determine how different they are. This value will depend on the practical application we deal with, but in order to standardize the energy values, a normalization process is carried out for a given set of shapes. For normalization, we first calculate the average of the energy values obtained when comparing two different images corresponding to the same key. Afterwards, we divide all the values in the table by this factor. In table 2, the normalized minimum energy values are shown for the comparisons of 9 different images corresponding to 3 keys (see Fig. 2), where $k_{n:m}$ corresponds to the m^{th} image of the n^{th} key (n^{th} key of the m^{th} row). Normalized energy values around or lower than 1 indicate a great similarity between the shapes which are compared, while those values which are much higher than 1 indicate that the contours correspond to clearly different shapes. When comparing the keys in the last row of Fig. 2 with those in the first two rows, the relationship described for reflected shapes is used. In order to decide whether we must use the direct or the reflected relationship, we compare the results for both situations and determine whether the keys are presented in the same or the reflected position, according to the option which generates a lower value.

3.5 Weighting Functions for Frequency Terms

Taking into account that the higher the order of the coefficient, the more sensitive it is to noise, we can weight the energy factors in such a way that the first



Fig. 2. Images of three different keys in different positions and showing both sides

Table 2. Normalized minimum energy values for keys in Fig. 2

$E_{\min}$	$k_{1:1}$	$k_{1:2}$	$k_{1:3}$	$k_{2:1}$	$k_{2:2}$	$k_{2:3}$	$k_{3:1}$	$k_{3:2}$	$k_{3:3}$
$k_{1:1}$	0.0000	1.1273	0.8219	3.5856	4.1306	4.1161	6.8020	5.9809	5.6455
$k_{1:2}$		0.0000	0.5607	5.4166	5.7934	5.7902	7.5247	6.4634	6.3026
$k_{1:3}$			0.0000	4.5853	5.2063	5.0216	7.4232	6.1860	6.0272
$k_{2:1}$				0.0000	0.9050	1.1134	8.6651	8.5584	8.0708
$k_{2:2}$					0.0000	1.0617	8.9780	6.7763	6.6304
$k_{2:3}$						0.0000	8.8944	8.9579	8.3986
$k_{3:1}$							0.0000	1.3734	1.2102
$k_{3:2}$								0.0000	0.8266
$k_{3:3}$									0.0000

coefficients are more significant than the last ones, as shown with the weighting function $w(\cdot)$ in equation (7).

$$E'(a) = \sum_{k=1}^{\frac{L}{2}} w\left(\frac{2k}{L}\right) \left(\left| \tilde{f}_k \right|^2 + \left| \tilde{g}_k \right|^2 - e^{i\frac{2\pi ka}{L}} \tilde{f}_k \tilde{g}_k^* - \left(e^{i\frac{2\pi ka}{L}} \tilde{f}_k \tilde{g}_k^* \right)^* \right) . \tag{7}$$

Different linear, quadratic and exponential weighting functions have been tested to reduce higher frequency factors while preserving the information contained in lower frequencies. In order to test how suitable a certain weighting function is for a given set of shapes, we have used the ratio between the lowest energy obtained for two images of different shapes and the highest energy for two images of the same shape.

The best ratios are observed when exponential functions of the form e^{-sx} are used, taking into account that the factor s in the exponent affects considerably the results. The ratios are improved as we increase the factor up to a maximum which is found around $s = 10$. From this value on, the ratio decreases and it is lower than 1 from $s = 50$ on, which prevents us from using these values for discrimination.

Table 3. Normalized minimum energy values for keys in Fig. 2 and weighting function $w(x) = e^{-10x}$

$E'_{\min}$	$k_{1:1}$	$k_{1:2}$	$k_{1:3}$	$k_{2:1}$	$k_{2:2}$	$k_{2:3}$	$k_{3:1}$	$k_{3:2}$	$k_{3:3}$
$k_{1:1}$	0.0000	1.1595	0.7754	4.8422	5.4602	4.8694	11.0473	9.0618	8.5987
$k_{1:2}$		0.0000	0.4007	8.2601	8.5063	7.0163	11.9956	9.2841	8.9991
$k_{1:3}$			0.0000	6.7835	6.8951	5.9262	11.8338	9.0418	8.7061
$k_{2:1}$				0.0000	0.8663	1.3553	14.1622	14.3641	13.0408
$k_{2:2}$					0.0000	1.0698	14.5646	12.3600	11.6192
$k_{2:3}$						0.0000	14.3818	15.8590	13.7275
$k_{3:1}$							0.0000	1.5318	1.1737
$k_{3:2}$								0.0000	0.6675
$k_{3:3}$									0.0000

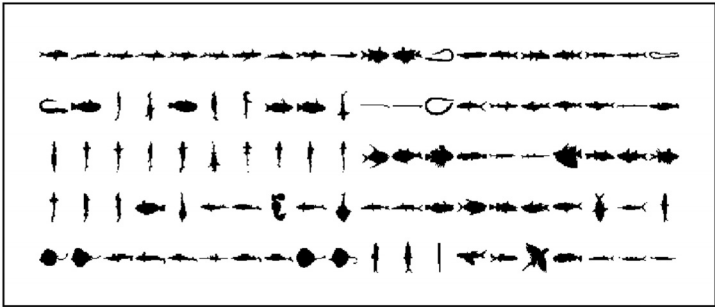


Fig. 3. 100 out of the 1000 pictures in the database

Table 3 shows the final values of normalized minimum energy for the keys in Fig. 2 when the factors are weighted as in equation (7), using $w(x) = e^{-10x}$ as weighting function. As observed when comparing tables 2 and 3, energy values increase significantly for images of different keys when the weighting function is used, which results in a clearer discrimination of the shapes. A reduction in the higher frequencies will also reduce the noise effects and facilitate the discrimination criteria. Nevertheless, the more detailed the forms, the higher the frequencies we will need to consider.

We have used a database containing 1000 pictures of different kinds of marine animals in such a way that, for a given picture of a fish, the most similar ones are selected. Some examples of these pictures are shown in Fig. 3. This fish database has been kindly provided to us by Professor Farzin Mokhtarian at the Centre for Vision, Speech, and Signal Processing of the University of Surrey, United Kingdom [6]. Of course, this classification is performed according to the shape of the silhouette and no other aspect, such as biological factors, is considered to determine which fishes are the most similar to the selected one. Some results can be observed in Fig. 4.

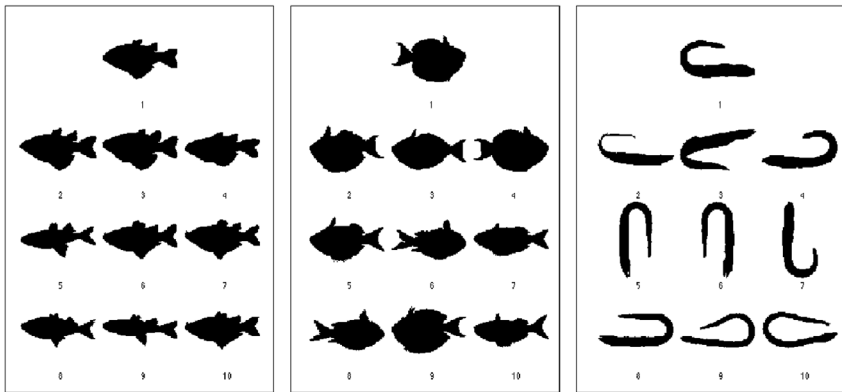


Fig. 4. Results of searching for similar shapes for three images in the database

3.6 Characterization of Partially Occluded Shapes

One of the most important problems when trying to identify an object is the fact that it may not be entirely visible in the scene we are working on. When identifying shapes, the object is not always completely available for its recognition, or several overlapping objects may be present in the image. This makes the possibility of occlusion arise in such a way that only some parts of the contour of each object are suitable to describe it.

Once we have obtained the orientation functions for two images where common parts are visible, we must determine exactly where these parts are located in both images. We must take into account that they may be continuous sections of the orientation functions or not, thus containing separate segments.

Firstly, we must extract the matching parts in both images, considering the Euclidean transformations, i.e. translations and rotations, which may have been produced. We assume that the scale is the same for both images. As many small common parts may be found and those straight lines which may appear, e.g. with pseudo-polygonal shapes, may be matched in several parts of the contour, we must select the longest one to fix the rotation which relates both pictures from the difference between the coupled points in both orientation functions. To select such segment, the difference between the values in the orientation functions of both pieces must be less than a certain threshold along the whole sub-signal.

If we want to extract other parts which also correspond to the same object, but which are not contiguous to the previous ones, the difference in the orientation must be approximately the same for all couples of segments, since we assume that the objects are rigid, as shown in Fig. 5. Besides, the translation from the rotated points of one of the contours to those in the other one must also be the same. When both constraints are satisfied, the segments are supposed to belong to the same contour, and thus, they can be considered as visible parts of the same object.

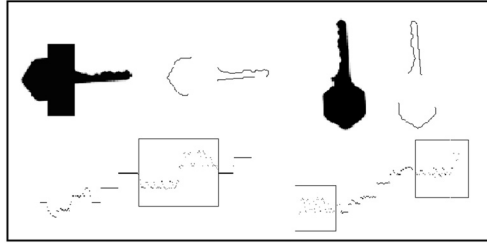


Fig. 5. Example of coincident parts in the orientation functions

4 Motion Analysis

If we try to analyze the motion of an object, several images must be compared. Global parameters can be extracted from the comparison of the object in different time instants. In fact, the position of the center gives information about the translation, the area covered by the object determines the changes in the size, and the shape of the object will help us decide the rotation angle. In order to process a sequence of frames, we must first study how to compare two of them in isolation.

Firstly, a translation may be required. It can be extracted from the centers of both coupled shapes by subtracting their coordinates. We can obtain the center (x_c, y_c) , as well as the translation T to be performed, from the position of the pixels which are covered by the object in the image.

Secondly, we must consider their sizes. As said before, orientation functions have been normalized to make comparisons. However, we must work with actual sizes, which can be extracted from the segmentation of the images in two regions, object and background. The square root of the quotient of both sizes determines the proportional scale S we must use.

Finally, the rotation to be performed with respect to the center can be identified from the mean values of both orientation functions. Nevertheless, a more accurate value for the rotation angle θ can be obtained as shown below, where (x_i^1, y_i^1) and (x_i^2, y_i^2) are the coupled points of the contour of both signals whose coordinates are calculated with respect to their respective centers:

$$\theta = \arctan \left(\frac{\sum_{i=1}^{L-1} (x_i^1 y_i^2 - y_i^1 x_i^2)}{\sum_{i=1}^{L-1} (x_i^1 x_i^2 + y_i^1 y_i^2)} \right) . \quad (8)$$

With these 3 transformations, we could adapt the position, orientation and distance of an object. A point (x, y) in the second object is transformed according to the parameters (T, S, θ) as follows:

$$(z, t) = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} ((x, y) + T - (x_c^2, y_c^2)) S + (x_c^2, y_c^2) - (x_c^2, y_c^2) + (x_c^2, y_c^2) . \quad (9)$$

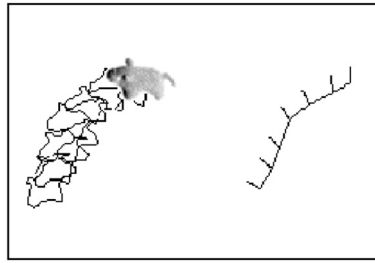


Fig. 6. Image sequence and temporal evolution. The trajectory of the object is shown on the right and the vectors represented on it indicate the changes in the orientation from the initial situation

A certain measure of the correctness of the association must be used. To this effect, we have selected the mean distance from the points in the second contour to those in the first contour, once the transformation has been carried out with the parameters extracted from the frames.

Figure 6 shows a real sequence of frames and the trajectory which has been extracted. On the left, the initial position of a small toy is shown, as well as the contours for the following frames. On the right, the trajectory is described indicating the changes in the orientation of the object with a vector whose angle corresponds to the rotation of the object and whose length reflects the size changes.

5 Conclusion

The present work proposes some new methods for shape characterization and motion analysis based on a new set of tools for characterizing edges. These are inspired on Newton Filters and allow an accurate processing of local information to build a global representation of a shape.

From this kind of basic filters, it is possible to build orientation functions which can clearly identify shapes and extract global information to fit objects into described patterns. We have used the discrete Fourier transform as fundamental tool in our analysis, which provides robust results and allows reducing the computational cost. Moreover, the introduction of a weighting function which affects the contribution of every term in the energy function allows regulating the frequencies which will be more relevant in the discrimination. The information extracted for a segment of an orientation function permits relating different parts of a sequence in such a way that we can associate the visible regions of an object when it is not completely accessible.

Furthermore, the fact that the information supplied by these mechanisms is larger and more accurate than that provided by simply detecting edges allows us to use them for a reliable motion analysis. In this case, the contours which are associated do not belong to different scenes, but to different instants in a video. Once the shapes have been analyzed and coupled, the transformations which

bring one of them to the other make it possible to extract motion parameters such as translation, rotation and scaling in time.

The numerical experiences are very promising and show a great coherence with the similarity measures and associations that humans use. In particular, we can even discriminate between shapes which are very similar from a perceptual point of view and follow the trajectory of a moving object.

References

1. Alemán-Flores, M., Álvarez-León, L., Moreno-Díaz jr., R.: Modified Newton Filters for Edge Orientation Estimation, Shape Representation and Motion Analysis. Cuadernos del Instituto Universitario de Ciencias y Tecnologías Cibernéticas, Universidad de Las Palmas de Gran Canaria **17** (2001) 1–30
2. Moreno-Díaz jr., R.: Computación Paralela y Distribuida: Relación Estructura-Función en Retinas. Tesis Doctoral (1993)
3. Loncaric, S.: A Survey of Shape Analysis Techniques. Pattern Recognition **31**(8) (1998) 983–1001
4. Zahn, C., Roskies, R.: Fourier Descriptors for Plane Closed Curves. Computer Graphics and Image Processing **21** (1972) 269–281
5. Rui, Y., She, A.C., Huang, T.S.: Modified Fourier Descriptors for Shape Representation - A Practical Approach. In Proceedings of the 1st Workshop on Image Databases and Multimedia Search (1996)
6. Mokhtarian, F.: Centre for Vision, Speech, and Signal Processing of the University of Surrey. <http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html>

Preprocessing Phase in the PIETSI Project (Prediction of Time Evolution Images Using Intelligent Systems)

J.L. Crespo, P. Bernardos, M.E. Zorrilla, and E. Mora

Department of Applied Mathematics and Computer Sciences,
University of Cantabria.

Avda. de los Castros s/n 39005 Santander. SPAIN

{crespoj, bernardp, zorrillm, morae}@unican.es

Abstract. We outline the PIETSI project, the core of which is an image prediction strategy, and discuss common preliminary image processing tasks that are relevant when facing problems such as: useless background, information overlapping (solvable if dissimilar coding is being used) and memory usage, which can be described as “marginal information efficiency”.

1 Presentation of Project

The work described in this paper forms one of the phases of the PIETSI research project. The main aim of this project is to predict the temporal evolution of physical processes or phenomena for which experimental information in image files is available.

A further aim is to obtain methodological results and software applications which can be of broad use for this type of problem. However, due to the vast range of possibilities and the convenience of working with information related to a real case which will allow the conclusions to be validated, some specific developments have been initiated, focusing on weather forecasting using synoptic map images. For this purpose, the University of Cantabria group has worked in collaboration with the EUVE (European Virtual Engineering) Technology Centre, and more specifically its Meteorological Centre, with its head office in Vitoria, Spain.

In short, the project could be said to have the following three different but complementary lines of action:

- Storage and management of multimedia information (in this case, images which represent the time evolution of a space distribution of values of physical magnitudes) by means of database techniques.
- Design and development of Intelligent Systems, (see, for instance, the possibilities of neural networks reviewed in [1]), for:
 1. Classification and indexing of images according to the information contained in them, (see [2] for instance).
 2. Detection of differentiated structures in an image.
 3. Prediction of images as a continuation of others, (see [3] for an approach based on segmentation and optical flow).

- Application to management of meteorological information, particularly predictive analysis, to the classification of types of weather and to the positioning of hydrometeorological space variables.

2 Considerations on Images to Be Used

The evolution of a phenomenon, the study of which is based on sequences of images, does not always begin with the images themselves. Thus, it often occurs that the images are used to facilitate the interpretation of information which is previously available; that is, each image is a graphic representation obtained from certain initial data which define it in some way. In the case of synoptic maps, for example, the isobars are built up from data provided by a set of meteorological centres.

In many cases, the image is then altered by an expert in order to increase its graphic information content. This is the case of the representation of cold, warm and hidden fronts in the synoptic maps used by the weather services. The final image thus contains the initial data highlighted and the information provided by the expert.

Furthermore, this image will normally contain information which is not relevant for the phenomenon to be represented, as occurs, in the case of weather maps, with parallel or meridian lines or the contours of the land over the sea. Thus, even before proceeding to store the image in the database, the elements of the image which are relevant for the phenomenon under analysis should be isolated, and those not under analysis cleared. In general, it will not be sufficient to decide which elements of the image are of interest, but rather to consider whether they are all to undergo the same analysis or need to be divided into groups to be studied separately.

Moreover, it should be borne in mind that the storage of an image might require a lot of space. This amount of information depends on the resolution, number of bands and depth. For the optimisation of the storage system, only that which is truly useful for the following process should be retained. For example: Is the original image in colour ? Is it necessary to retain it ? Is the number of colours significant ? Can shades of colour be grouped together without a loss of information ? Is the density of points indispensable ? How much can be reduced without affecting the performance of the subsequent processes ?

In the preliminary operations to discern between the information which is “of interest” and that which is not, one should work with the original level of detail (resolution, colours, size, ...) reducing it only when moving on to the version to be stored. In this way, the possible loss of information in a previous stage is avoided.

For a general background of common image processing operations, see reference [4].

2.1 What Is of Interest and What Is Not

As mentioned above, there may be some elements of the image which are not of interest for the subsequent analysis. These may be fixed, which means that for the computer the image will have a “systematic anomaly”, or variable, in which case they will have “random anomalies”. The elimination of non-relevant fixed information can be performed by taking two images and generating from them a set of points to be

eliminated, which will be practically those which are common to both images. In fact, in this process, some spurious points are generated, being those that are common but which have truly useful information and only coincide by chance. This can be avoided by taking only those points which are common to all or at least several images.

Figure 1 shows the image of a usual synoptic map which contains information that is not relevant for weather forecasting. In particular, the non-relevant fixed information to be eliminated is the following: the parallel and meridian lines, the contours of the land over the sea and the upper left and lower right captions in the image.

The operations to be carried out to eliminate the points which are common to two images may depend on the type of operators available on the programs. One example is:

$$(\text{image 1 AND } (\text{image 1 XOR image 2})). \quad (1)$$

The well known XOR logical operation executed between the two images has the same effect as to subtract them, hence, except for common points, only the isobars and fronts of the two images remain. Now, executing the AND logical operation between this previous result and image 1, the common points present in these two images are obtained, namely, the isobars and fronts of image 1. Figure 2 shows the image exposed in Figure 1 after receiving the “cleaning” treatment mentioned above.

It must be pointed out that this and subsequent image operations are related to white lines on black background, while all figures are presented as black lines on white background, for readability.

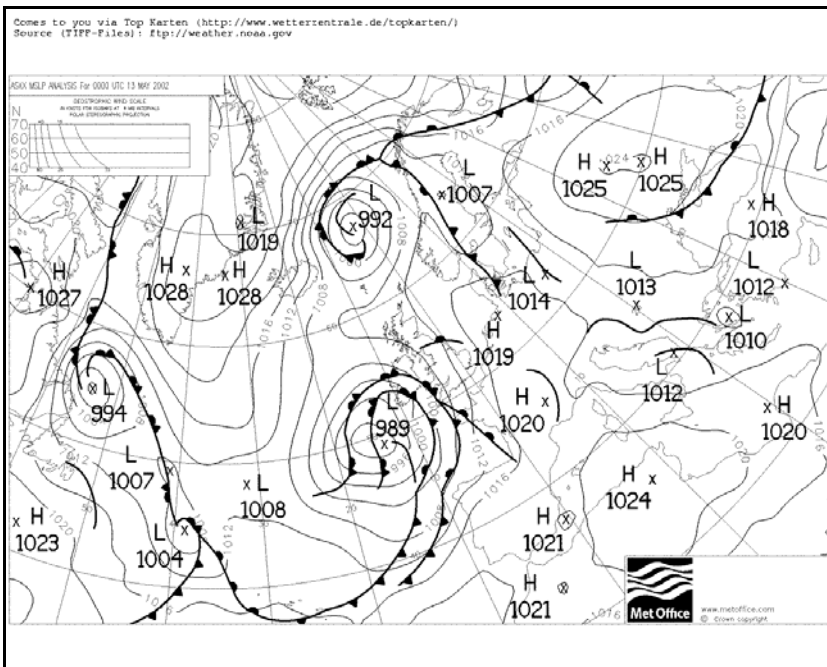


Fig. 1. Synoptic map with all its information.

Dilation (Erosion (image)) AND image. (3)

It is easy to see that an appropriate erosion of the image in Figure 2 will cause the thin lines to disappear and the thick ones to narrow; then by dilating them afterwards (which is an opening operation), and executing the AND logical operator with the starting figure, it is possible to isolate the information from thick lines of the image in Figure 2, as is shown in Figure 3.

In the same way, the creation of an image in which the thin lines have been separated from Figure 2, can be performed by means of the operation:

Dilation (Erosion (image)) XOR image. (4)

The procedure is similar as before but now, Figure 2 must be subtracted from its dilated eroded image, in order to obtain the isobars. Figure 4 shows the isolated information from thin lines of the image in Figure 2.

2.3 Information for Storage

When the image is intended for human visualisation, it requires a degree of quality which makes it not only intelligible, but also aesthetically pleasing, or at least above certain minima in this respect. However, the computer may be content with intelligibility alone, so that the size may be reduced. The question to be answered is: how much efficiency is achieved by the system when a certain amount of information (resolution, depth, bands) is added?. It can be seen as a “marginal information efficiency problem”. The final solution involves using an information amount smaller than that of the original image. This reduction will have a significant impact if the number of images to be stored is high.

In this sense, from a prior knowledge of the image, decisions can be made about the number of bands, resolution, size required and depth.

Most image acquisition devices output files in colour format (usually RGB 8 bit files). Generally, subsequent processing takes place in grey scale, so that a first step in the preprocessing phase consists in reducing the number of bands, which although conceptually trivial, has a great impact in the amount of memory needed.

Regarding the resolution to be adopted, it is also possible to perform tests to verify whether, after a reduction, the image continues to be intelligible; this leads to a process of “manual” selection of the desired degree of reduction. In some cases, it may be possible to recognise automatically the appropriate degree of reduction by generating an original size image taken from the reduced one and a subsequent study of their correlation. Figures 5 and 6 show how the information from the above example in Figures 3 and 4 continues to be intelligible after reducing resolution by 50% on each axis.

In this particular project, it will be appropriate to find the area around Spain that has a significant influence upon its weather. Thus, it is obvious that the isobars and the front lines far from Spain in the synoptic maps can be eliminated. For example, taking a region of interest downsized 50% on each axis, would mean a saving in space of 75%. Hence, by focusing on the lower right quadrant from Figures 3 and 4, the images shown in Figures 7 and 8, respectively, appear.

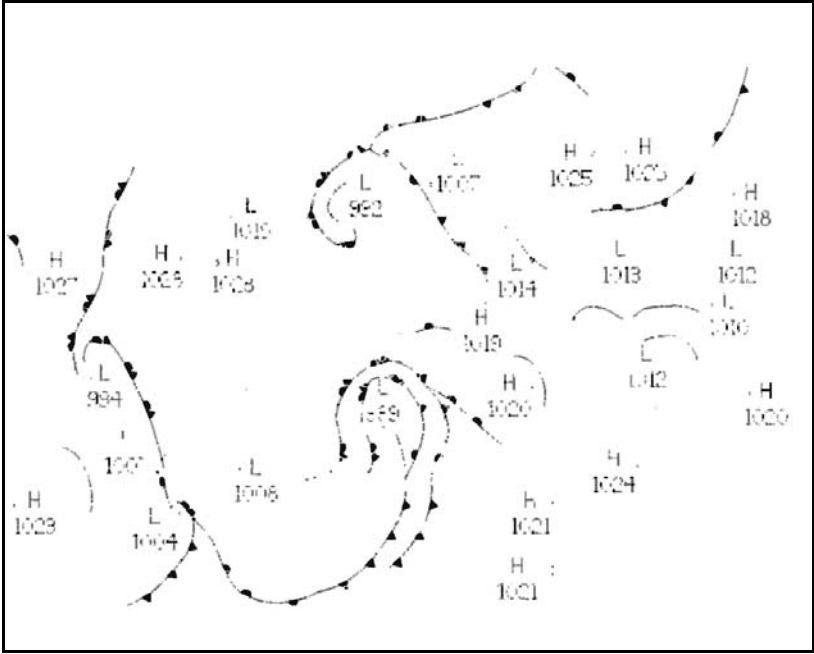


Fig. 3. Image of Figure 2 after the process of selection of thick lines.

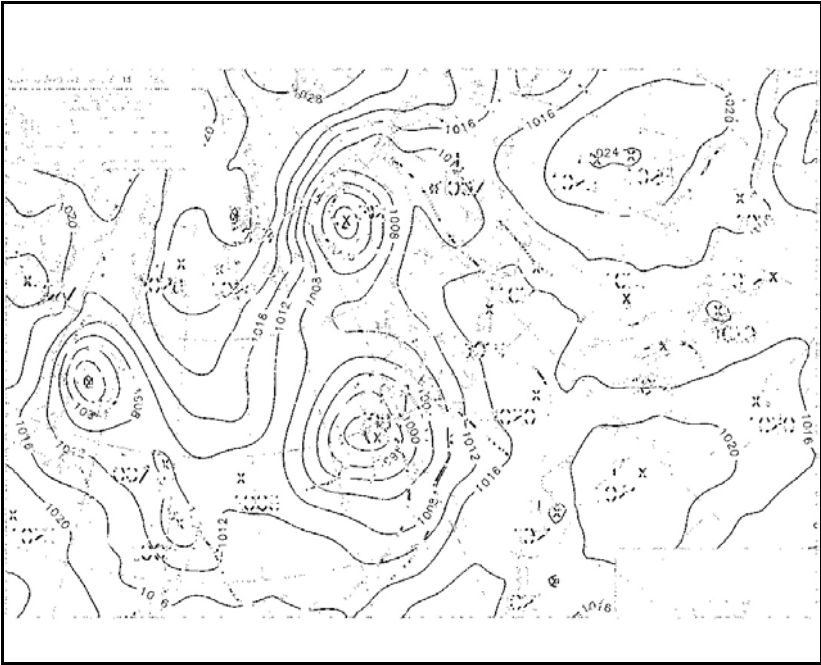


Fig. 4. Image of Figure 2 after the process of selection of thin lines.

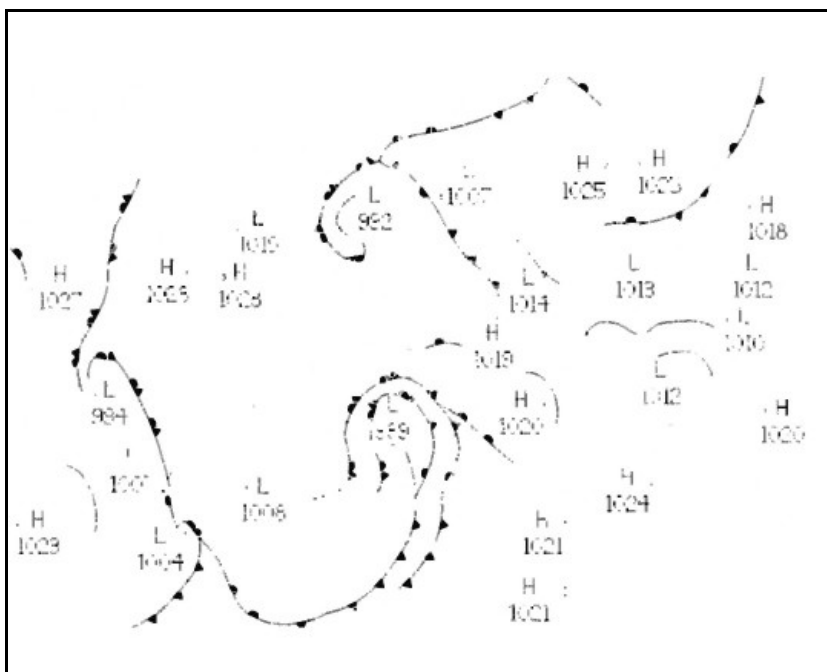


Fig. 5. Image of Figure 3 after reducing resolution by 50%.

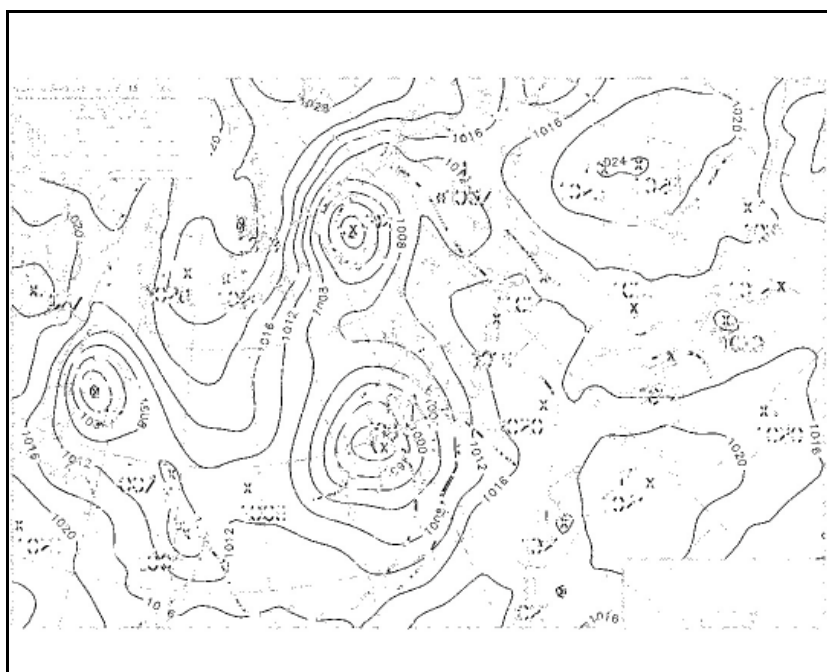


Fig. 6. Image of Figure 4 after reducing resolution by 50%.

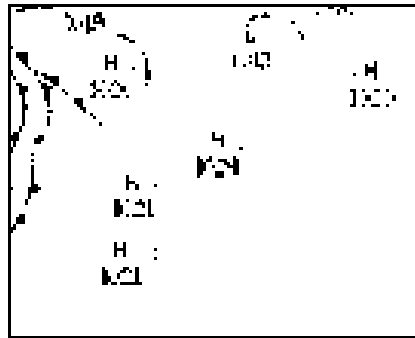


Fig. 7. Image of Figure 3 corresponding to the quadrant from the lower right side.

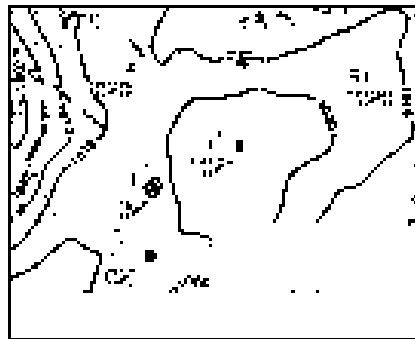


Fig. 8. Image of Figure 4 corresponding to the quadrant from the lower right side.

Moreover, these two images are binary; that is to say, the different levels of grey of each pixel are eliminated, their being reduced a white or black attribute.

Summarizing, the reduction in the number of bands means dividing by three the required memory, a 50% reduction in the resolution implies dividing by four, a 50% area of interest on each axis also dividing by four and, finally, converting an image in a grey 8 bit scale to a binary one means dividing by eight. In spite of the original image becoming two, the final saving in memory, in this case, is around 99.5%.

3 Conclusions and Future Work

The PIETSI project is related mainly with an image prediction strategy. To accomplish this aim, an adequate knowledge of the images is required and some preprocessing image tasks have to be carried out in order to extract the information of interest. Moreover, given that the number of images to be processed and stored is high, it is an important question to reduce the memory needed for storing them. Thus, in this work, a general presentation of the project is made and the first part is reviewed in detail, that is to say, the image preprocessing phase. Common preliminary tasks in this sense, such as useless background removing, information separation and memory usage are discussed.

We feel it is interesting to indicate that the next stage of the project is to find in the images a specific area of interest resulting from our knowledge of the meaning of the images. The second step will be to define these images with a few variables such as the density of the lines and their orientations in several zones, in order to regard a classification of the images according to the information included in them. The objective is to distinguish between the main different types of information which are present in those images so as to be able to use only a reduced set of images in the prediction task.

Acknowledgments. The authors are deeply grateful to the Spanish Interministerial Board for Science and Technology (Comisión Interministerial de Ciencia y Tecnología CICYT) which supports Project TIC2002-01306, as a part of which this paper has been produced. They would also like to thank the EUVE (European Virtual Engineering) Technology Centre for their collaboration.

References

1. Egmont-Petersen, M. et al. Image processing with neural networks – a review. Pattern recognition. 2001.
2. Breiteneder, C.; Eidenberger, H. Performance-optimized feature ordering for Content-Based Image Retrieval. Proc. European Signal Processing Conference. Tampere, 2000.
3. G. Bors, I. Pitas. Prediction and Tracking of Moving Objects in Image Sequences. IEEE Trans. on Image Processing, vol. 9, no. 8. 2000.
4. R. C. González y R. E. Woods. Digital Image Processing. Prentice-Hall, 2002.

Devices to Preserve Watermark Security in Image Printing and Scanning

Josef Scharinger

Johannes Kepler University,
Institute of Systems Sciences,
4040 Linz, Austria
`js@cast.uni-linz.ac.at`

Abstract. In this contribution we specify novel digital watermark embedding and detection algorithms that offer remarkable robustness in situations when an image is printed and later on scanned in again. Since this process can significantly affect image resolution and is commonly not done with perfect accuracy, image scaling potentially involving different scaling factors in horizontal and vertical direction must be dealt with. As will be demonstrated in the contribution, the concept specified herein outperforms commercial digital image watermarking systems with respect to scale-invariance and is capable of detecting watermarks in images that have been printed and scanned. Accordingly it can be expected that these findings will be particularly useful in copyright protection schemes that involve various media transitions from digital to analog and back.

1 Introduction

Copyright protection for digital images is a complex issue. One has to find a way to attach a signature to the digital image that does not visually interfere with image quality and is additionally very robust against attempts to remove the copyright information. Inherently this robustness requirement excludes any attempt to attach the copyright information as some kind of header information to the image. Simply cutting off that header block removes the intellectual property rights code without doing any damage to the pixel data and thus leaving the image perfectly intact for an intruder. Therefore any potentially successful approach to robust image copyright protection has to embed the copyright code directly into the pixel data in a very complex way that makes it impossible for an attacker to remove the code from the image data without significantly deteriorating image quality. If that condition is met removal attempts may be possible in principle but the result is of inferior quality and useless for the intruder.

A promising approach that has attracted a lot of attention recently is based on so-called digital watermarks. Informally speaking, a digital watermark is some imperceptible embedded control code carrying information related to the intellectual property rights of the data. This may be utilized not just to limit the number of copies that can be made, but also provides means to control the distribution.

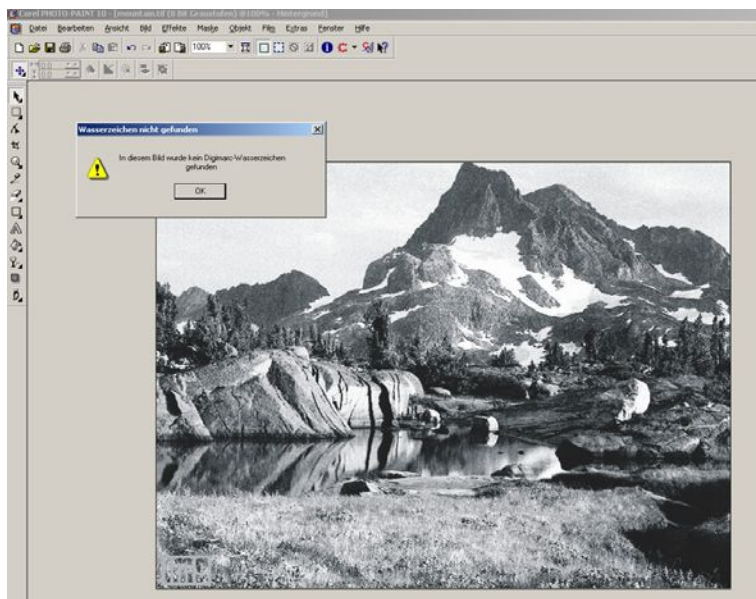


Fig. 1. Often watermarks are not robust with regard to minor modifications

In just one decade remarkable progress has been made within the research community as well as in the commercial branch making available a large variety of different products for the purpose of digital watermarking [4,11,9,3,7,10]. On the other hand, systems developed for the purpose of (at least partially) removing watermarks [6,20] have shown quite effective against many of the commercially available watermarks and emphasize that the field of digital image watermarking is far from being fully explored.

As mentioned before, an essential topic in digital watermarking is robustness against modifications made to the watermarked digital product. If these modifications do not cause significantly change the image then the watermark should still be detectable in the modified image. However, in current watermarking systems even minor modifications that do not noticeably decrease image quality cause great problems for the watermark detector. Just consider Fig. 1. Here a test image has been watermarked using a commercial image watermarking system [3] at maximum embedding strength and then rescaled in horizontal direction by 130% and in vertical direction by 129%. As shown¹ in Fig. 1 even this visually negligible alteration causes the detector to fail!

Motivated by this observation this contribution specifies novel embedding and detection algorithms that offer much higher robustness in situations when an image is printed out and later on scanned in again. Since this process can significantly affect image resolution and is commonly not done with perfect accuracy, image scaling potentially involving different scaling factors in horizontal and vertical direction had to be taken into consideration. Deducing from many

¹ The alert message popped up translates to "No watermark found in this image".

experiments conducted it is well justified to assume that numerous copyright protection applications which involve media transitions from digital to analog and back could benefit a lot from the algorithms described herein.

2 Conventional DSSS Digital Image Watermarking

Most of the commercially available systems for digital image watermarking are based on ideas known from spread spectrum radio communications [5,15]. In spread spectrum communications, one transmits a narrowband signal over a much larger bandwidth such that the signal energy present in any single frequency is undetectable [2]. This allows the signal reception even if there is interference on some frequencies [1].

Spread-spectrum techniques have been used since the mid-fifties in the military domain because of their anti-jamming and low-probability-of-intercept properties [14], their applicability to image watermarking has only been noticed recently [18,17]. Since then, a large number of systems have been proposed based on this technique, including many of the commercially most successful ones. Although there are many variants of spread-spectrum communications, we will focus on *Direct-Sequence Spread Spectrum* (DSSS) as the method most useful for application in digital image watermarking.

In DSSS, the bandwidth of a data sequence is opened using modulation, multiplying the narrow band data sequence waveform $b(t)$ by a broad band pseudo-noise (PN) sequence signal $c(t)$ to generate the transmitted signal [21]

$$m(t) = c(t)b(t). \quad (1)$$

The product signal will have a spectrum equal to the convolution of the two separate spectra and which will therefore be nearly the same as the wide band PN signal spectrum. Thus the PN sequence is called a spreading code.

The received signal $r(t)$ consists of the transmitted signal plus some additive interference $i(t)$ and is thus given by

$$r(t) = m(t) + i(t) = c(t)b(t) + i(t). \quad (2)$$

To recover the original signal the received signal is input to a demodulator consisting of a multiplier followed by an integrator. The multiplier is supplied with an exact copy of the PN sequence $c(t)$ delivering an output given by

$$z(t) = c^2(t)b(t) + c(t)i(t) = b(t) + c(t)i(t), \quad (3)$$

using the simplification $c^2(t) = 1$ for all t . The original signal is multiplied twice by the PN sequence and is recovered to its narrow band form while the interference is multiplied once and is spread in spectrum at the multiplier output. An integrator after the multiplier can therefore filter out the original signal $b(t)$ from the interference $i(t)$.

Concepts from DSSS are carried over to applications in digital image watermarking in a straightforward manner. A descriptive exposition how this can be

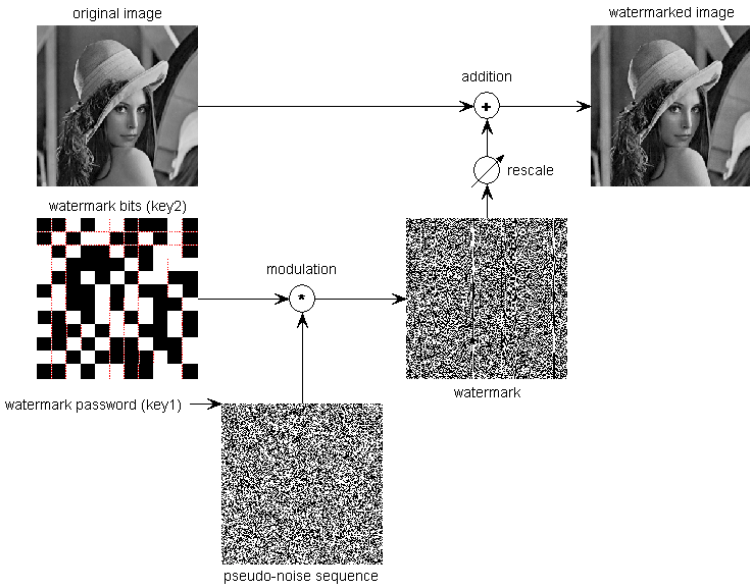


Fig. 2. Direct Sequence Spread Spectrum (DSSS) watermark embedding

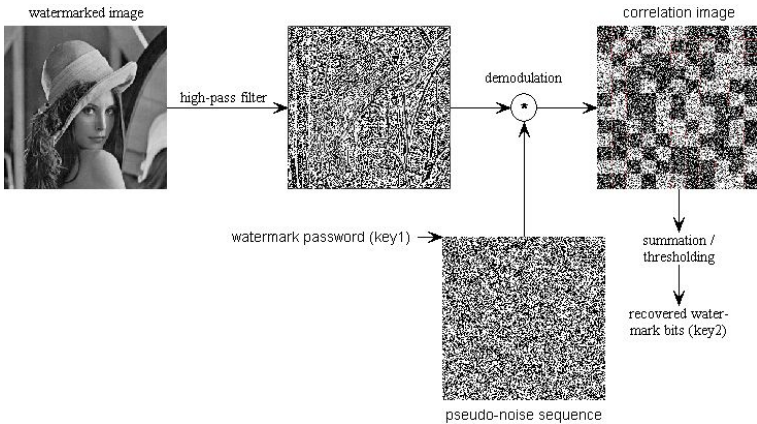


Fig. 3. Direct Sequence Spread Spectrum (DSSS) watermark detection

achieved is found e.g. in [8] and similarly in [13]; to illustrate the principle we will closely follow along these lines.

Figure 2 illustrates a simple, straightforward example of spread spectrum watermarking. The watermark bits (**key2**) to be embedded are spread to fill an image of the same size as the image to be watermarked. The spread information bits are then modulated with a cryptographically secure pseudo-noise (PN) signal keyed by watermarking key **key1**, scaled according to perceptual criteria, and added to the image in a pixel-wise fashion.

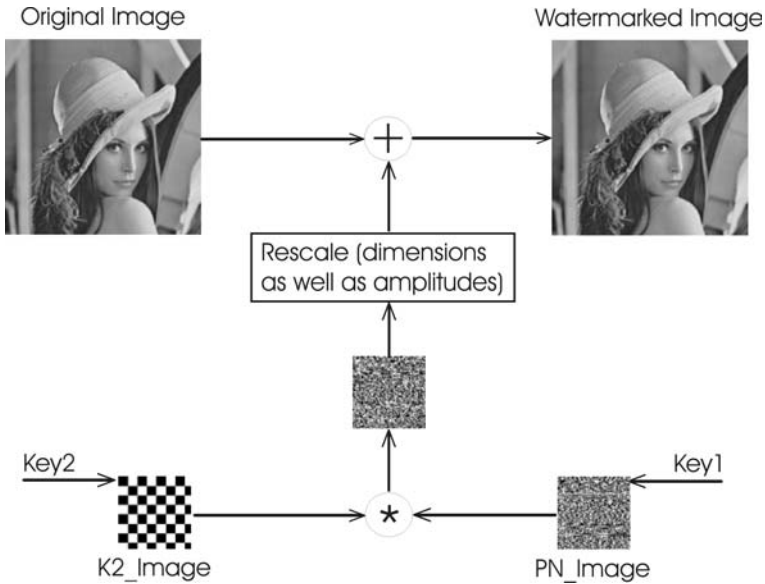


Fig. 4. Scale-invariant watermark embedding

Figure 3 illustrates the corresponding watermark detector based on the principle of a correlation receiver (matched filter). In order to reduce cross-talk between the image and the watermark, a pre-filter is applied first to remove low frequencies corresponding largely to the image from the signal received. If the original un-watermarked image is available to the detector, the signal-to-noise ratio can greatly be increased if the filtering is replaced by subtraction of the original image. After filtering/subtraction, the signal is demodulated using exactly the same PN signal used for watermark embedding. The samples of the correlation signal are summed for each embedded watermark bit, and a threshold application finally yields the output bits.

3 Novel Approach for Scale-Invariant Watermarking

As pointed out in the introduction and illustrated in Fig. 1, DSSS watermarks are extremely sensitive to geometric distortions. To improve on this situation we have designed a novel approach for robust digital image watermark embedding and detection which should prove extremely helpful in situations when watermarked images are printed and scanned in later on. The embedding algorithm is depicted in Fig. 4 whereas the corresponding detector is shown in Fig. 5.

The basic idea is quite simple. DSSS watermarks are high-frequency signals by definition. The watermarks we embed and detect are much more of a low-frequency nature. Therefore, as shown in Fig. 4, watermarks are defined on a nominal scale $n = N \times N$ much smaller than the image dimensions, scaled in amplitude and dimensions and finally added to the original image.

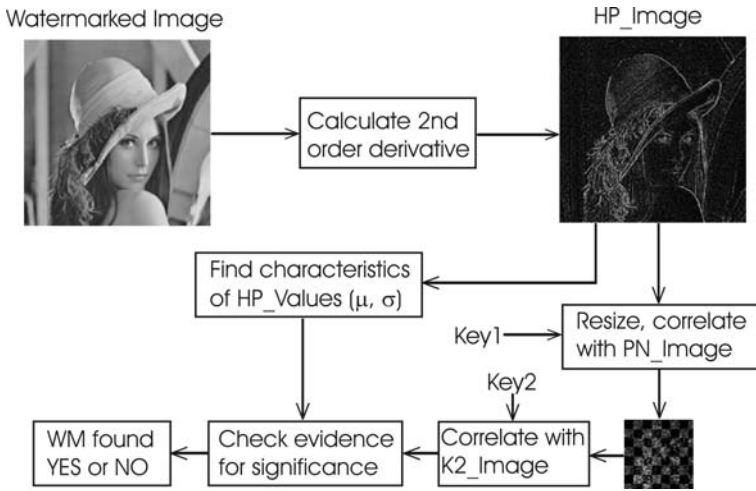


Fig. 5. Scale-invariant watermark detection

Due to the low-frequency character of the embedded watermark, the detector (see Fig. 5) is significantly more robust to geometric distortions. After eliminating the influence of the original image as much as possible by clever filtering (LOG, DOG [12]), values are sub-sampled to the nominal scale and correlated with sequences derived from respective keys. This results in a measure of evidence that a watermark has been embedded at a specific nominal scale using specific watermarking keys. Note that the original image is not needed for detection at all so an intelligent peripheral scanning device just needs knowledge of keys and nominal scale used in the watermark embedding process!

How can we assess if the evidence observed is significant. Reconsider the Normal Distribution $N(\mu, \sigma)$ characterized by mean μ and standard deviation σ given by

$$N(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \quad (4)$$

According to the Central Limit Theorem [16], the following holds. Given a set $X_1, X_2, \dots, X_n$ of independent, identically distributed stochastic variables. Suppose each X_i has mean $E(X_i) = \mu$ and variance $Var(X_i) = \sigma^2 < \infty$. Define $Y_n = \sum_{i=1}^n X_i$. Then

$$\lim_{n \rightarrow \infty} \frac{Y_n - n\mu}{\sqrt{n}\sigma} \quad (5)$$

has a standard normal $N(0, 1)$ distribution. Particularly we get $\mu_{Y_n} = n\mu$ and $\sigma_{Y_n} = \sqrt{n}\sigma$.

Images are commonly considered to be non-stationary signals and samples taken at neighboring positions are usually highly correlated and thus far from independent, so the assumptions needed in the central limit theorem (Eq. 5) do not really hold in our context. Despite this shortcoming, summing up 2nd derivative values as depicted in Fig. 5 approximates a normal distribution (Eq. 4) reason-



Fig. 6. Original image *mountain*

ably well and an assessment of the evidence measured that works amazingly fine in practice can be developed as follows.

Under the assumption that no watermark is present, it is known [16] that 68% percent of observations fall within $n\mu \pm \sqrt{n}\sigma$, 95% fall within $n\mu \pm 2\sqrt{n}\sigma$ and 99,7% fall within $n\mu \pm 3\sqrt{n}\sigma$. Thus if we observe an evidence value larger than $3\sqrt{n}\sigma$ (μ is always very close to zero in this context!), we can be extremely confident that this happened not by chance, but delivers a really significant hint that a watermark is embedded in the image under consideration.

4 Experimental Results

In this section we give experimental results justifying the claim that our approach for robust scale-invariant watermarking can offer reliable watermark detection even in situations when a watermarked image is first printed and scanned in later on, a crucial requirement when attempting to realize intelligent image scanners including copyright enforcement functionality.

In Fig. 6 test image *mountain* taken from the set of test images provided by the *University of Waterloo* [19] is shown. As shown in Fig. 1, use of a commercial image watermarking system [3] and re-scaling in horizontal direction by 130% and in vertical direction by 129% causes that system to fail.

The approach introduced in this contribution does much better. In Fig. 7 an image carrying 3 watermarks produced with our approach is shown whereby embedding strength is specified in Tab. 1, left columns. Robustness to re-scaling becomes obvious in the middle columns of Tab. 1 where almost no reduction of the evidence to $\sqrt{n}\sigma$ ratio is observed.

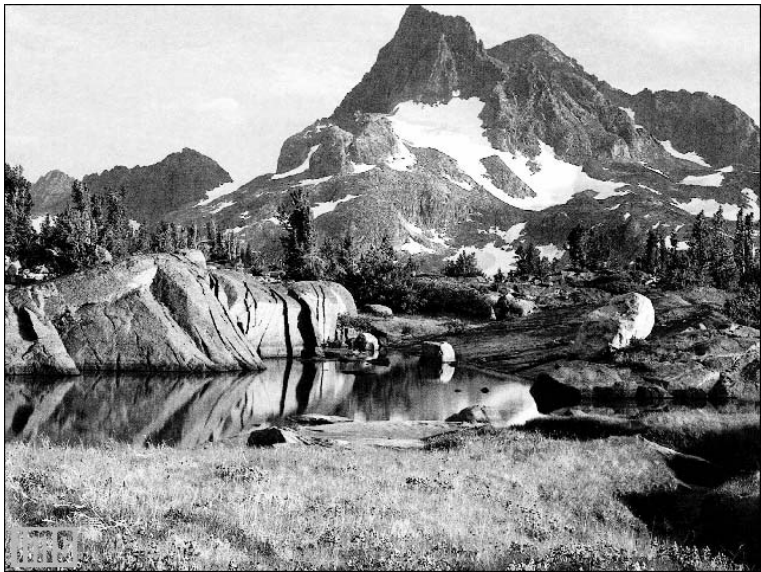


Fig. 7. Watermarked image carrying 3 different watermarks

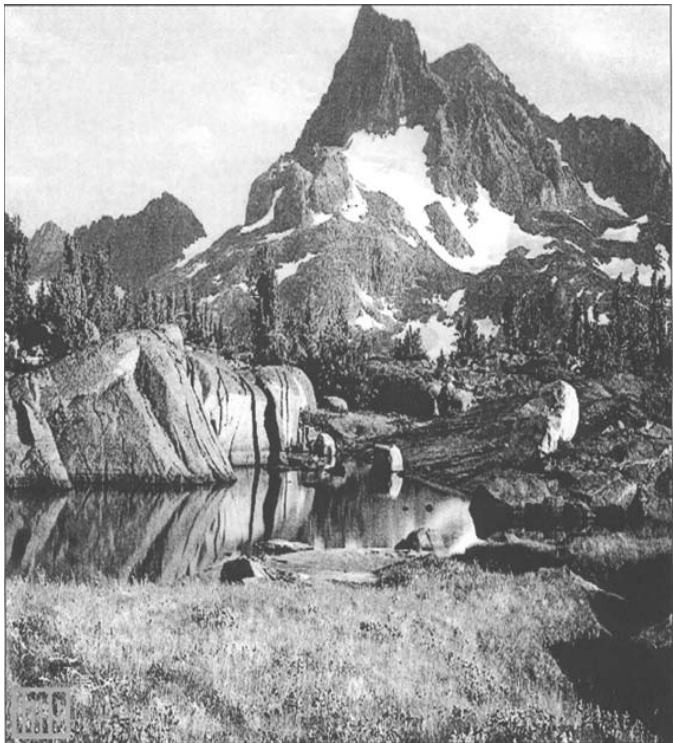


Fig. 8. Re-scaled, printed and re-scanned watermarked image

Table 1. Significance of watermarks detected

	watermarked image (640×480)		scaled to 576×634		printed, scanned at 566×626	
nominal size n	evidence	$\sqrt{n}\sigma$	evidence	$\sqrt{n}\sigma$	evidence	$\sqrt{n}\sigma$
128	13,66	3,86	10,23	2,96	6,30	1,94
256	19,50	1,93	15,24	1,48	6,19	0,97
512	20,83	0,97	14,41	0,74	4,39	0,48

But our approach offers more than just robustness against re-scaling. Consider Fig. 8. Here the watermarked image was first printed using an *HP LaserJet* and that hard-copy then scanned in with a pretty old *UMAX SuperVista*. Image quality suffers quite a lot, but detection evidence given in the right two columns of Tab. 1 is still beyond $3\sqrt{n}\sigma$ and therefore highly significant.

5 Conclusion

In this contribution we have specified novel digital watermark embedding and detection algorithms that can offer remarkable robustness when an image is printed out and later on scanned in again. Usually this process changes image resolution and is not performed with perfect accuracy, so image scaling potentially involving different scaling factors in horizontal and vertical direction had to be taken into consideration.

As has been demonstrated in the paper, the concept specified herein outperforms commercial digital image watermarking systems with respect to scale-invariance and is capable of detecting watermarks in images that have been printed and scanned. Therefore it can well be expected that these findings will be particularly useful in copyright protection schemes that involve various media transitions form digital to analog and back.

References

1. W. Bender, D. Gruhl, N. Morimoto, and A. Lu. Techniques for data hiding. *IBM Systems Journal*, 35(3&4), 1996.
2. I. Cox, J. Kilian, F.T Leighton, and T. Shamoan. Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing*, 6:1673–1687, December 1997.
3. Digimarc Corporation. Picture Marc. available at <http://www.digimarc.com>.
4. Digital Information Commodities Exchange (DICE). Argent. available at <http://digital-watermark.com>.
5. R. Dixon. *Spread Spectrum Systems*. John Wiley and Sons, New York, 1984.
6. Fabien A.P. Petitcolas and Markus G. Kuhn. StirMark 2.3 watermark robustness testing software. available at <http://www.cl.cam.ac.uk/~fapp2/watermarking/image.watermarking/stirmark/>.
7. Fraunhofer Center for Research in Computer Graphics (CRCG). Syscop. available at <http://syscop.igd.fhg.de>.

8. F. Hartung, J. Su, and B. Girod. Spread spectrum watermarking: Malicious attacks and counterattacks. In *Security and Watermarking of Multimedia Contents, Proc. SPIE 3657*, January 1999.
9. IBM, Howard Sachar. IBM Digital Library System. contact sachar@watson.ibm.com.
10. Ingemar J. Cox and Joe Kilian and Tom Leighton and Talal Shamoon. A secure robust watermark for multimedia. In *Information Hiding: First International Workshop, Volume 1174 of Lecture Notes in Computer Science*, pages 183–206, 1996.
11. Intellectual Protocols2 (IP2). Copysight. available at <http://www.ip2.com>.
12. Anil Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.
13. Martin Kutter. Performance improvement of spread spectrum based image watermarking schemes through m-ary modulation. In *Workshop on Information Hiding, Lecture Notes in Computer Science, volume 1768*, pages 238–250, 1999.
14. Fabien A.P. Petitcolas, Ross J. Anderson, and Markus G. Kuhn. Attacks on copyright marking systems. In *Second Workshop on Information Hiding, vol. 1525 of Lecture Notes in Computer Science*, pages 218–238, April 1998.
15. R.L. Pickholtz, D.L. Schilling, and L.B. Milstein. Theory of spread spectrum communications – a tutorial. *IEEE Transactions on Communications*, 30(5):855–884, May 1982.
16. Sheldon M. Ross. *Introduction to Probability and Statistics for Engineers and Scientists*. Academic Press, 2 edition, 2000.
17. R.V. Schyndel, A. Tirkel, and C. Osborne. A digital watermark. In *Proceedings of the IEEE International Conference on Image Processing*, pages 86–90, November 1994.
18. A.Z. Tirkel, G.A. Rankin, R.G. van Schyndel, W.J. Ho, N.R.A. Mee, and C.F. Osborne. Electronic watermark. In *Dicta-93*, pages 666–672, 1993.
19. University of Waterloo, Ontario, Canada. The Waterloo Fractal Compression Project. Waterloo Repertoire GreySet2. Available at <http://links.uwaterloo.ca/greyset2.base.html>.
20. Valentin Lacambre (a.k.a *Altern*). unZign watermark removal software. available at <http://altern.org/watermark/>.
21. John A.R. Williams. Spread spectrum modulation. Aston University, available at <http://www.eeap.aston.ac.uk/teltec/tutorials/>, 1999.

Author Index

- Abalde, C. 220, 253
Affenzeller, Michael 384
Alberola-López, Carlos 506, 597
Albertos, P. 139
Albrecht, Rudolf F. 10
Alemán-Flores, Miguel 639
Alghisi Manganello, Elisa 482
Alonso-Jiménez, José A. 115
Álvarez, Pedro 231
Álvarez-León, Luis 616, 639
- Bañares, José A. 231
Baños, Karina 616
Barberá, E. 139
Benítez, I.J. 139
Bergasa, Luis Miguel 302
Bernardos, P. 651
Bhalerao, Abhir 573
Bienvenido, J. Fernando 627
Bistarelli, S. 415
Blanco Ferro, A. 279
Blasio, Gabriel de 494
Blauth Menezes, Paulo 62, 243
Borrego-Díaz, Joaquín 115
Bosch, Alfonso 185
Bresson, Xavier 585
Brun, Anders 518
Bubnicki, Z. 38
- Calderón, A.J. 337
Campos, M. 208
Cárdenes, Rubén 542, 597
Cervesato, I. 415
Češka, Milan 265
Chen, Y.Q. 337
Corigliano, Pietro 302
Costa, Simone André da 243
Crespo, J.L. 651
Cuenca, Carmelo 616
Cuisenaire, Olivier 585
Cull, Paul 349
- Delgado, Ana E. 427, 460
Deriche, R. 530
Di Nardo, Elvira 394
Díez, J.L. 139
- Doussin, Marie-Hélène 302
- Esclarín, Julio 616
- Faugeras, Olivier 552
Fernández, Cristina 26
Fernández, Miguel A. 427
Fernández-Caballero, Antonio 427
Flores, Ramón 302, 326
Flores-Parra, Isabel M. 627
Freire Brañas, E. 279
Freire Nistal, J.L. 220, 279
Fuzitaki, Claudio Naoto 243
- García, Miguel Ángel 326
García, R. 291
García-Cabrera, Lina 196
García-Tizón, J. 220
Gea Megías, M. 50
Giorno, Virginia 360
González, A. 208
González, C. 291
González-Careaga, Rafaela 448
González Cobas, Juan David 19
Gudbjartsson, Daniel 372
Guil, Francisco 185
Gulías, Victor M. 220, 279
Gutiérrez-Naranjo, Miguel A. 115
- Hagmann, Patric 585
Hansen, Jens A. 372
Haša, Luděk 265
Hsu, Tai 349
- Ingólfssdóttir, Anna 372
- Jablonski, Andrzej 174
Jacak, Witold 163
Johnsen, Jacob 372
Jonasson, Lisa 585
- Kalwa, Joerg 302
Klempous, Ryszard 174
Knudsen, John 372
Knutsson, Hans 518, 564

- Leibovic, K.N. 471
 Lenzini, G. 415
 Licznarski, Benedykt 174
 Llamas, B. 208
 Lope, Javier de 436, 448
 López, María T. 427
 López Brugos, José Antonio 19

 Macías, Elsa M. 542
 Madsen, Anders L. 302
 Magdalena, Luis 302
 Marangoni, R. 415
 Maravall, Darío 314, 436, 448
 Marín, Roque 185, 208
 Mariño, C. 253
 Martín, Jacinto 151
 Martín-Fernández, Marcos 506
 Martinelli, F. 415
 Mata, Eloy J. 231
 Medina-Medina, Nuria 196
 Menárguez, M. 208
 Meuli, Reto 585
 Mira, José 427, 460
 Miró, Josep 74
 Miró-Juliá, Margaret 92
 Molina-Ortiz, Fernando 196
 Monje, C.A. 337
 Mora, E. 651
 Moraes Claudio, Dalcidio 62
 Moreno-Díaz, Arminda 151
 Moreno-Díaz, Roberto 494
 Moreno-Díaz jr., Roberto 471, 639
 Muro-Medrano, Pedro R. 231

 Naranjo, J.E. 291
 Németh, Gábor 10
 Nobile, Amelia G. 360, 394
 Norman, R. 404

 Ocaña, Manuel 302

 Palma, J. 208
 Parets-Llorca, José 196
 Park, Hae-Jeong 518
 Patricio, Miguel Ángel 314
 Pedro, T. de 291
 Penas, M. 253
 Penedo, M.G. 253
 Pereira Machado, Júlio 243
 Perrier, Michel 302
 Pichler, Franz 1
 Pirozzi, Enrica 360, 394

 Pröll, Karin 163

 Quesada-Arencibia, A. 471

 Resconi, Germano 104, 482
 Reviejo, J. 291
 Reyes-Aldasoro, Constantino Carlos 573
 Ricciardi, Luigi M. 360, 394
 Ríos Insua, David 151
 Rodríguez Almendros, M.L. 50
 Rodríguez-Florida, M.A. 597
 Rodríguez Fórtiz, M.J. 50
 Rodríguez-Rodríguez, J.C. 471
 Roland, Damien 302
 Rovaris, Eduardo 607
 Rozenblit, Jerzy 163
 Rubio, Julio 231
 Ruiz-Alzola, Juan 542, 597, 607

 San José-Estépar, Raul 506
 Sánchez, J. 220
 Sánchez, Javier 616
 Santana, Jose Aurelio 542, 607
 Santos, Matilde 26
 Santos Leal, Liara Aparecida dos 62
 Sarrió, M. 139
 Scharinger, Josef 660
 Schwaninger, Markus 127
 Shankland, C. 404
 Shimogawa, Takuhei 83
 Sotelo, Miguel Ángel 302, 326
 Suárez, Eduardo 607
 Suárez, J.I. 337

 Taboada, M.J. 460
 Thiran, Jean-Philippe 585
 Thirion, Bertrand 552
 Tschumperlé, D. 530
 Túnez, Samuel 185

 Vieira Toscani, Laura 62
 Vinagre, B.M. 337
 Virto, Miguel A. 151
 Vojnar, Tomáš 265

 Wagner, Stefan 384
 Warfield, Simon K. 542
 Westin, Carl-Fredrik 506, 518, 564, 597, 607

 Zarraonandia, Telmo 448
 Zorrilla, M.E. 651