

# Delay Differential Equations

Balakumar Balachandran • Tamás Kalmár-Nagy  
David E. Gilsinn  
Editors

# Delay Differential Equations

Recent Advances and New Directions

*Editors*

Balakumar Balachandran  
Department Mechanical Engineering  
University of Maryland  
2181 Glenn L. Martin Hall  
College Park, MD 20742  
USA  
balab@umd.edu

Tamás Kalmár-Nagy  
Department of Aerospace Engineering  
Texas A&M University  
609C H. R. Bright Building  
College Station, TX 77843  
USA  
kalmarnagy@aero.tamu.edu

David E. Gilsinn  
Mathematical & Computational Sciences  
Division  
National Institute of Standards  
and Technology (NIST)  
100 Bureau Drive, Stop 8910  
Gaithersburg, MD 20899  
USA  
david.gilsinn@nist.gov

ISBN: 978-0-387-85594-3

e-ISBN: 978-0-387-85595-0

DOI: 10.1007/978-0-387-85595-0

Library of Congress Control Number: 2008935618

© Springer Science+Business Media, LLC 2009

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed on acid-free paper

springer.com

# Preface

Delay differential equations (DDEs) are important in many areas of engineering and science. The aim of this book is to bring together contributions from leading experts on the theory and applications of functional and DDEs. The editors have strived throughout to point out the interdisciplinary nature of these contributions. For example, advances in stability analysis, computational techniques (symbolic and numerical), and results from many otherwise disconnected fields (automotive engineering, manufacturing, neuroscience, and control theory) have been included in this book. The different contributions are not intended as a comprehensive survey of the many areas in which functional and DDEs have been useful, but rather, they are meant to help a reader bridge the gap between theoretical work and applications and extend the results and methods to other fields.

This book contains eleven chapters. These chapters have been organized into five groups, covering control systems and stability analysis methods, Hopf bifurcations and center manifold analysis, numerical computations of DDE solutions, neural systems, and stochastic DDEs.

The first group on control systems and stability analysis methods consists of five chapters. In the first three chapters, delay effects in control systems are addressed. In Chap. 1, Hongfei Li and Keqin Gu provide an exposition of the Lyapunov–Krasovskii functional approach for studying the stability of equilibrium points and other solutions of coupled differential–difference equations with multiple delays. Such coupled differential–difference systems occur in the context of control processes, propagation in electrical lines, and fluid and gaseous flows through pipes. They show that retarded and neutral systems can be considered as special cases of coupled differential–difference systems. For linear systems, the authors show how a quadratic Lyapunov functional can be used to convert the stability problem to a set of linear matrix inequalities, which can be numerically studied using MATLAB and other software. This stability treatment is also used in Chap. 2, where Wenjuan Jiang, Alexandre Kruszewski, Jean-Pierre Richard, and Armand Toguyeni consider the stability and control of networked systems. They discuss the different sources of delays in these systems, including network-induced hardware-dependent delays and packet losses. The challenging aspect of the control of these networked systems is

that the delays are varying, unpredictable, and can also be unbounded. Control of a remote system through the Internet is examined; controller and observer designs are examined on the basis of available theoretical results for systems with time-varying delays, and it is demonstrated that a gain scheduling strategy can be used to achieve an exponential stabilization of this system. An experimental platform is also proposed to explore the control architecture. The discussion presented in this chapter is valid for networks operated through the Internet as well as the Ethernet. In Chap. 3, Mrdjan Jankovic and Ilya Kolmanovsky examine automotive powertrain control and discuss the different possible sources of delays in these systems and treat idle speed control and air-to-fuel ratio control in gasoline engines. Through experiments and simulations, the authors bring forth the power of applying different control tools for robust and stochastic stabilization of these systems. Various models, controller and observer designs are presented in this work. It is shown that a careful consideration of time delays is important for enhancing the fuel economy, emissions, and drivability of automotive vehicles.

Rounding up the first group, in Chaps. 4 and 5, the use of discretization methods for stability analysis of delayed systems with periodic coefficients as well as constant or periodically varying delays is illustrated. Eric Butcher and Brian Mann elaborate methods based on Chebyshev polynomial and expansion methods and temporal finite element analysis in Chap. 4. They consider different applications, including optimal and delayed state feedback control of systems with time periodic delays and chatter vibrations in machine-tool systems. In Chap. 5, Xinhua Long, Tamás Insperger, and Balakumar Balachandran consider the semidiscretization method at length, use a variety of examples, and show how stability information for periodic solutions of autonomous and nonautonomous delay differential systems with constant as well as time periodic delays can be obtained by constructing an approximation to the infinite dimensional monodromy matrix. As in Chap. 4, stability charts are used to identify stable and unstable regions in the considered parameter space. The possibilities for different local bifurcations of periodic solutions, including cyclic-fold, period-doubling, and Hopf bifurcations, are also discussed in this chapter.

The second group consists of Chaps. 6–8. In these chapters, the basis for analytical and symbolic bifurcation studies in nonlinear systems with constant time delays is carefully introduced. David Gilsinn presents a detailed discussion of how center manifold and normal form analysis commonly used for studying local bifurcations in ordinary differential systems can be used for examining Hopf bifurcation of fixed points of delay differential systems in Chap. 6. It is shown how such an analysis is important for determining the nature of a bifurcation, that is, whether it is subcritical or supercritical. Machine-tool chatter is used as an illustrative example. In a complementary effort, in Chap. 7, Siming Zhao and Tamás Kalmár-Nagy rigorously examine the existence of a Hopf bifurcation of the zero solution of the delayed Liénard equation. They also illustrate how the continuation-based DDE-Biftool and MATLAB's DDE-23 integrator can be used to numerically investigate such systems. Sue Ann Campbell demonstrates how symbolic algebra packages such as MAPLE can be used to carry out center manifold analysis in Chap. 8. As in Chap. 6, first, the

basis for ordinary differential systems is discussed for a reader to relate the approach used for delay differential systems with the one followed for ordinary differential equations.

The third group includes Chap. 9, in which Larry Shampine and Skip Thompson focus on numerical solutions of delay differential systems. They make comparisons with existing methods commonly used for ordinary differential systems and elaborate on how the popular Runge–Kutta methods can be extended to delay differential systems. They detail at length as to how algorithms written in MATLAB and Fortran 90/95 can be used to study solutions of a range of delay differential systems.

The fourth group consists of Chap. 10. In this chapter, Qishao Lu, Qingyun Wang, and Xia Shi study the effects of delays in coupled neuronal systems. They present different models including the models of Chay, Hodgkin–Huxley, FitzHugh–Nagamo, and Hindmarsh–Rose and show how an appropriate consideration of the delays in these systems is important to uncover the rich dynamics exhibited by these systems. Observed behaviors include bifurcations, bursting, chaos, and wave dynamics. The implications of these phenomena for synchronization of neurons are discussed, and the importance of considering them for characterizing the neural activity of the central nervous system is brought forth in this chapter.

Chapter eleven fills out the fifth group. In this chapter, Toru Ohira and John Milton examine the interplay between delay and noise in the context of postural actions and demonstrate how an appropriate formulation of the delayed random walk problem can serve as an alternate as well as complimentary approach for examining stochastic delay differential systems such as the delayed Langevin equation. The findings of this chapter are relevant for understanding feedback control mechanisms that are common in physiology as well as in other systems from biology and economics.

Through the eleven contributions, the editors have attempted to provide a glimpse of the recent advances in the area of delay differential systems and new directions being pursued in this area. While it is recognized that the material presented in this edited book is by no means complete, it is our sincere hope that this book will allow a reader to appreciate the ubiquitous presence of delays in various systems, learn about new and exciting areas, and develop a mastery of tools available for treating delay differential systems.

We thank the different contributors for their spirited participation in this endeavor and Ms. Elaine Tham and Ms. Lauren Danahy of Springer for helping us see this collaborative effort to fruition.

College Park, MD  
College Station, TX  
Gaithersburg, MD

*Balakumar Balachandran*  
*Tamás Kalmár-Nagy*  
*David E. Gilsinn*

# Contents

<b>1</b>	<b>Lyapunov–Krasovskii Functional Approach for Coupled Differential-Difference Equations with Multiple Delays . . . . .</b>	<b>1</b>
	Hongfei Li and Keqin Gu	
1.1	Introduction . . . . .	1
1.2	Coupled Differential-Functional Equations . . . . .	5
1.3	Stability of Continuous Time Difference Equations . . . . .	10
1.4	Linear Coupled Differential-Difference Equations . . . . .	13
1.4.1	Quadratic Lyapunov-Krasovskii Functional . . . . .	13
1.4.2	Discretization . . . . .	16
1.5	Discussion and Examples . . . . .	23
1.6	Concluding Remarks . . . . .	27
	References . . . . .	28
<b>2</b>	<b>Networked Control and Observation for Master–Slave Systems . . . . .</b>	<b>31</b>
	Wenjuan Jiang, Alexandre Kruszewski, Jean-Pierre Richard, and Armand Toguyeni	
2.1	Introduction . . . . .	31
2.2	Exponential Stability of a Remote System Controlled Through Internet . . . . .	33
2.2.1	The Three Delay Sources . . . . .	33
2.2.2	Transmission and Receipt of the Control Data . . . . .	34
2.2.3	Problem Formulation and Preliminaries . . . . .	34
2.2.4	Observer Design . . . . .	35
2.2.5	Control Design . . . . .	36
2.2.6	Global Stability of the Remote System . . . . .	37
2.3	Architecture of the Global Control System . . . . .	38
2.3.1	Features of the Remote System . . . . .	38
2.3.2	Synchronization of the Time Clocks . . . . .	38
2.3.3	Transmission and Receipt of The Control Data . . . . .	39
2.3.4	The Structure of the Master . . . . .	40
2.3.5	The Structure of the Slave . . . . .	41
2.3.6	Experimental Study . . . . .	42

2.4	Performance Enhancement by a Gain Scheduling Strategy . . . . .	45
2.4.1	Effects of Time-Delay on the Performance and the System Stability . . . . .	45
2.4.2	Uniform Stability with Gain Scheduling . . . . .	46
2.4.3	Gain Scheduling Experiments . . . . .	48
2.4.4	Result of Remote Experiment . . . . .	48
2.5	Conclusion . . . . .	51
	References . . . . .	52
<b>3</b>	<b>Developments in Control of Time-Delay Systems for Automotive Powertrain Applications . . . . .</b>	<b>55</b>
	Mrdjan Jankovic and Ilya Kolmanovsky	
3.1	Introduction . . . . .	55
3.2	Idle Speed Control . . . . .	56
3.2.1	A Model for ISC . . . . .	58
3.2.2	Treating ISC Using Tools for Time-Delay Systems . . . . .	60
3.3	Air-to-Fuel Ratio Control . . . . .	63
3.3.1	Feed-gas Air-to-Fuel Ratio Control Block Diagram . . . . .	65
3.3.2	Modeling . . . . .	66
3.3.3	Constructing Stability Charts and Gain Scheduling . . . . .	69
3.3.4	Using Robust and Stochastic Stabilization Tools for Time-Delay Systems . . . . .	71
3.3.5	Air-to-Fuel Ratio Control Using a Switching HEGO Sensor . . . . .	75
3.4	Observer Design for a Diesel Engine Model with Time Delay . . . . .	78
3.4.1	Diesel Engine Model . . . . .	78
3.4.2	Observer Design . . . . .	82
3.4.3	Numerical Results . . . . .	85
3.4.4	Summary and Discussion . . . . .	87
3.5	Concluding Remarks . . . . .	88
	References . . . . .	90
<b>4</b>	<b>Stability Analysis and Control of Linear Periodic Delayed Systems Using Chebyshev and Temporal Finite Element Methods . . . . .</b>	<b>93</b>
	Eric Butcher and Brian Mann	
4.1	Introduction . . . . .	93
4.2	Stability of Autonomous and Periodic DDEs . . . . .	96
4.3	Temporal Finite Element Analysis . . . . .	97
4.3.1	Application to a Scalar Autonomous DDE . . . . .	97
4.3.2	TFEA Approach Generalization . . . . .	101
4.3.3	Application to Time-Periodic DDEs . . . . .	102
4.4	Chebyshev Polynomial Expansion and Collocation . . . . .	104
4.4.1	Expansion in Chebyshev Polynomials . . . . .	105
4.4.2	Estimating the Number of Polynomials . . . . .	108
4.4.3	Chebyshev Collocation Method . . . . .	109
4.5	Application to Milling Stability . . . . .	113

4.6	Control of Periodic Systems with Delay using Chebyshev Polynomials . . . . .	116
4.6.1	Variation of Parameters Formulation . . . . .	116
4.6.2	Finite Horizon Optimal Control via Quadratic Cost Function . . . . .	117
4.6.3	Optimal Control Using Convergence Conditions . . . . .	118
4.6.4	Example: Optimal Control of a Delayed Mathieu Equation . . . . .	119
4.6.5	Delayed State Feedback Control . . . . .	120
4.6.6	Example: Delayed State Feedback Control of the Delayed Mathieu Equation . . . . .	122
4.7	Discussion of Chebyshev and TFEA Approaches . . . . .	123
	References . . . . .	126
<b>5</b>	<b>Systems with Periodic Coefficients and Periodically Varying Delays: Semidiscretization-Based Stability Analysis . . . . .</b>	<b>131</b>
	Xinhua Long, Tamás Insperger, and Balakumar Balachandran	
5.1	Introduction . . . . .	131
5.2	Stability Analysis of Systems with Periodically Varying Delays . . . . .	133
5.3	Approximation of the Monodromy Matrix by using the Semidiscretization Method . . . . .	135
5.4	Applications . . . . .	138
5.4.1	Scalar Delay Differential Equation with Periodic Coefficient and One Delay . . . . .	138
5.4.2	Scalar Autonomous Delay Differential Equation with Two Delays . . . . .	141
5.4.3	Damped and Delayed Mathieu Equation . . . . .	142
5.4.4	CSS Milling Process: Nonautonomous DDE with Time-Periodic Delays . . . . .	144
5.4.5	VSS Milling Process: Nonautonomous DDE with Time-Periodic Delay . . . . .	148
5.5	Closure . . . . .	151
	References . . . . .	152
<b>6</b>	<b>Bifurcations, Center Manifolds, and Periodic Solutions . . . . .</b>	<b>155</b>
	David E. Gilsinn	
6.1	Background . . . . .	155
6.2	Decomposing Ordinary Differential Equations Using Adjoints . . . . .	160
6.2.1	Step 1: Form the Vector Equation . . . . .	160
6.2.2	Step 2: Define the Adjoint Equation . . . . .	161
6.2.3	Step 3: Define a Natural Inner Product by way of an Adjoint . . . . .	161
6.2.4	Step 4: Get the Critical Eigenvalues . . . . .	162
6.2.5	Step 5: Apply Orthogonal Decomposition . . . . .	163
6.3	An Example Application in Ordinary Differential Equations . . . . .	164
6.3.1	Step 1: Form the Vector Equation . . . . .	165

6.3.2	Step 2: Define the Adjoint Equation	165
6.3.3	Step 3: Define a Natural Inner Product by Way of an Adjoint	165
6.3.4	Step 4: Get the Critical Eigenvalues	166
6.3.5	Step 5: Apply Orthogonal Decomposition	166
6.4	Delay Differential Equations as Operator Equations	167
6.4.1	Step 1: Form the Operator Equation	167
6.4.2	Step 2: Define an Adjoint Operator	169
6.4.3	Step 3: Define a Natural Inner Product by Way of an Adjoint Operator	170
6.4.4	Step 4: Get the Critical Eigenvalues	171
6.4.5	Step 5: Apply Orthogonal Decomposition	172
6.5	A Machine Tool DDE Example: Part 1	176
6.5.1	Step 1: Form the Operator Equation	176
6.5.2	Step 2: Define the Adjoint Operator	178
6.5.3	Step 3: Define a Natural Inner Product by Way of an Adjoint	179
6.5.4	Step 4: Get the Critical Eigenvalues	179
6.5.5	Step 5: Apply Orthogonal Decomposition	185
6.6	Computing the Bifurcated Periodic Solution on the Center Manifold	186
6.6.1	Step 1: Compute the Center Manifold Form	187
6.6.2	Step 2: Develop the Normal Form on the Center Manifold	188
6.6.3	Step 3: Form the Periodic Solution on the Center Manifold	189
6.7	A Machine Tool DDE Example: Part 2	190
6.7.1	Step 1: Compute the Center Manifold Form	190
6.7.2	Step 2: Develop the Normal Form on the Center Manifold	194
6.7.3	Step 3: Form the Periodic Solution on the Center Manifold	195
6.8	Simulation Results	196
	References	201
<b>7</b>	<b>Center Manifold Analysis of the Delayed Liénard Equation</b>	<b>203</b>
	Siming Zhao and Tamás Kalmár-Nagy	
7.1	Introduction	203
7.2	Linear Stability Analysis	204
7.3	Operator Differential Equation Formulation	206
7.4	Center Manifold Reduction	209
7.5	Hopf Bifurcation Analysis	212
7.6	Numerical Results	213
7.7	Hopf Bifurcation in the Sunflower Equation	214
7.8	Concluding Remarks	218
	References	218

<b>8</b>	<b>Calculating Center Manifolds for Delay Differential Equations Using Maple™</b>	221
	Sue Ann Campbell	
8.1	Introduction	221
8.2	Theory	222
8.2.1	Linearization	224
8.2.2	Nonlinear Equation	227
8.3	Application	230
8.4	Discussion	240
	References	242
<b>9</b>	<b>Numerical Solution of Delay Differential Equations</b>	245
	Larry F. Shampine and Sylvester Thompson	
9.1	Introduction	245
9.2	DDEs are not ODEs	247
9.3	Numerical Methods and Software Issues	252
9.3.1	Explicit Runge–Kutta Methods	253
9.3.2	Error Estimation and Control	255
9.3.3	Event Location	256
9.3.4	Software Issues	257
9.4	Examples	258
9.4.1	El-Niño Southern Oscillation Variability Model	259
9.4.2	Rocking Suitcase	260
9.4.3	Time-Dependent DDE with Impulses	268
9.5	Conclusion	269
9.6	Further Reading	269
	References	269
<b>10</b>	<b>Effects of Time Delay on Synchronization and Firing Patterns in Coupled Neuronal Systems</b>	273
	Qishao Lu, Qingyun Wang, and Xia Shi	
10.1	Introduction	273
10.2	Basic Concepts	276
10.2.1	Firing Patterns of a Single Neuron	276
10.2.2	Synchronization	277
10.3	Synchronization and Firing Patterns in Electrically Coupled Neurons with Time Delay	280
10.4	Synchronization and Firing Patterns in Coupled Neurons with Delayed Inhibitory Synapses	283
10.5	Synchronization and Firing Patterns in Coupled Neurons with Delayed Excitatory Synapses	289
10.6	Delay Effect on Multistability and Spatiotemporal Dynamics of Coupled Neuronal Activity [14]	292
10.7	Closure	297
	References	303

<b>11</b>	<b>Delayed Random Walks: Investigating the Interplay Between Delay and Noise</b>	<b>305</b>
	Toru Ohira and John Milton	
11.1	Introduction	305
11.2	Simple Random Walk	307
11.2.1	Probability Distribution Function	309
11.2.2	Variance	312
11.2.3	Fokker–Planck Equation	312
11.2.4	Auto-correlation Function: Special Case	313
11.2.5	Auto-correlation Function: General Case	315
11.3	Random Walks on a Quadratic Potential	316
11.3.1	Auto-correlation Function: Ehrenfest Random Walk	318
11.3.2	Auto-correlation Function: Langevin Equation	320
11.4	Delayed Random Walks	320
11.4.1	Delayed Fokker–Planck Equation	321
11.4.2	Auto-correlation Function: Delayed Random Walk	322
11.4.3	Auto-correlation Function: Delayed Langevin Equation	324
11.5	Postural Sway	326
11.5.1	Transient Auto-correlation Function	329
11.5.2	Balance Control with Positive Feedback	330
11.6	Concluding Remarks	332
	References	332
<b>Index</b>		<b>337</b>

# Contributors

**B. Balachandran** Department of Mechanical Engineering, University of Maryland, College Park, MD 20742, USA, balab@umd.edu

**Eric Butcher** Department of Mechanical and Aerospace Engineering, New Mexico State University, Las Cruces, NM 88003, USA, eab@nmsu.edu

**Sue Ann Campbell** Department of Applied Mathematics, University of Waterloo, Waterloo, ON N2L, Canada

Centre for Nonlinear Dynamics in Physiology and Medicine, McGill University, Montreal, QC H3A 2T5, Canada, 3G1 sacampbell@uwaterloo.ca

**David E. Gilsinn** National Institute of Standards and Technology, 100 Bureau Drive, Stop 8910, Gaithersburg, MD 20899-8910, USA, dgilsinn@nist.gov

**Keqin Gu** Department of Mechanical and Industrial Engineering, Southern Illinois University Edwardsville, Edwardsville, IL 62026-1805, USA, kgu@siue.edu

**T. Insperger** Department of Applied Mechanics, Budapest University of Technology and Economics, Budapest H-1521, Hungary, Hungary inspi@mm.bme.hu

**Mrdjan Jankovic** Ford Research and Advanced Engineering, Dearborn, MI, USA, mjankov1@ford.com

**Wenjuan Jiang** LAGIS CNRS UMR 8146, Ecole Centrale de Lille, BP 48, 59651 Villeneuve d'Ascq, Cedex, France, wenjuan.jiang@ec-lille.fr

**Tamás Kalmár-Nagy** Department of Aerospace Engineering, Texas A&M University, College Station, TX 77843, USA, kalmarnagy@aeromail.tamu.edu

**Ilya Kolmanovsky** Ford Research and Advanced Engineering, Dearborn, MI, USA, ikolmano@ford.com

**Alexandre Kruszewski** LAGIS CNRS UMR 8146, Ecole Centrale de Lille, BP 48, 59651 Villeneuve d'Ascq, Cedex, France, Alexandre.Kruszewski@ec-lille.fr

**Hongfei Li** Department of Mathematics, Yulin College, Yulin City, Shaanxi Province 719000, P.R. China, [lhf8165@163.com](mailto:lhf8165@163.com)

**X.-H. Long** The State Key Lab of Mechanical System and Vibration, Shanghai Jiaotong University, Shanghai 200240, P.R. China, [xhlong@sjtu.edu.cn](mailto:xhlong@sjtu.edu.cn)

**Qishao Lu** School of Science, Beijing University of Aeronautics and Astronautics, Beijing 100191, China, [qishaolu@hotmail.com](mailto:qishaolu@hotmail.com)

**Brian Mann** Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708, USA, [brian.mann@duke.edu](mailto:brian.mann@duke.edu)

**John Milton** Joint Science Department, The Claremont Colleges, Claremont, CA, USA, [jmilton@jsd.claremont.edu](mailto:jmilton@jsd.claremont.edu)

**Toru Ohira** Sony Computer Science Laboratories, Inc., Tokyo, JAPAN 141-0022, [ohira@csl.sony.cp.jp](mailto:ohira@csl.sony.cp.jp)

**Jean-Pierre Richard** LAGIS CNRS UMR 8146, Ecole Centrale de Lille, BP 48, 59651 Villeneuve d'Ascq, Cedex, France Equipe-Projet ALIEN, INRIA (l'Institut National de Recherche en Informatique et en Automatique), [jean-pierre.richard@ec-lille.fr](mailto:jean-pierre.richard@ec-lille.fr)

**L.F. Shampine** Mathematics Department, Southern Methodist University, Dallas, TX 75275, USA, [shampine@smu.edu](mailto:shampine@smu.edu)

**Xia Shi** School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China, [berthashi@yahoo.com.cn](mailto:berthashi@yahoo.com.cn)

**S. Thompson** Department of Mathematics and Statistics, Radford University, Radford, VA 24142, USA, [thompson@radford.edu](mailto:thompson@radford.edu)

**Armand Toguyeni** LAGIS CNRS UMR 8146, Ecole Centrale de Lille, BP 48, 59651 Villeneuve d'Ascq, Cedex, France, [armand.toguyeni@ec-lille.fr](mailto:armand.toguyeni@ec-lille.fr)

**Qingyun Wang** School of Statistics and Mathematics, Inner Mongolia Finance and Economics College, Huhhot 010051, China, [nmqingyun@163.com](mailto:nmqingyun@163.com)

**Siming Zhao** Department of Aerospace Engineering, Texas A&M University, College Station, TX 77843, USA, [s0z0239@aeromail.tamu.edu](mailto:s0z0239@aeromail.tamu.edu)

## Chapter 1

# Lyapunov–Krasovskii Functional Approach for Coupled Differential-Difference Equations with Multiple Delays

Hongfei Li and Keqin Gu

**Abstract** Coupled differential-difference equations (coupled DDEs) represent a very general class of time-delay systems. Indeed, traditional DDEs of retarded or neutral type, as well as singular systems, can all be considered as special cases of coupled DDEs. The coupled DDE formulation is especially effective when a system has a large number of state variables, but only a few of them involve time delays. In this chapter, the stability of such systems is studied by using a Lyapunov-Krasovskii functional method. For linear systems, a quadratic Lyapunov-Krasovskii functional is discretized to reduce the stability problem to a set of linear matrix inequalities for which effective numerical algorithms are available, and widely implemented in such software packages as MATLAB.

**Keywords:** Coupled differential-difference equations · Time delay · Lyapunov-Krasovskii functional · Linear matrix inequality

## 1.1 Introduction

Coupled differential-difference equations (coupled DDEs)

$$\dot{x}(t) = f(t, x(t), y(t - r_1), y(t - r_2), \dots, y(t - r_K)), \quad (1.1)$$

$$y(t) = g(t, x(t), y(t - r_1), y(t - r_2), \dots, y(t - r_K)), \quad (1.2)$$

have received substantial attention from researchers since the early 1970s. Another seemingly more general form of coupled DDEs often studied in the literature is

$$\dot{x}(t) = f(t, x(t), x_d(t), y_d(t)), \quad (1.3)$$

$$y(t) = g(t, x(t), x_d(t), y_d(t)), \quad (1.4)$$

where

$$\begin{aligned} x_d(t) &= (x(t-r_1), x(t-r_2), \dots, x(t-r_K)), \\ y_d(t) &= (y(t-r_1), y(t-r_2), \dots, y(t-r_K)). \end{aligned}$$

Early motivation for studying such systems arose from some systems described by hyperbolic partial differential equations, with main examples from lossless propagation model of electrical lines and gas, steam, and water pipes [2, 23, 26, 29, 36, 44, 45, 47]. To illustrate the typical process, consider a lossless transmission line studied in [2] described by the partial differential equations

$$L \frac{\partial i}{\partial t} = -\frac{\partial v}{\partial x}, \quad (1.5)$$

$$C \frac{\partial v}{\partial t} = -\frac{\partial i}{\partial x}, \quad (1.6)$$

for  $0 \leq x \leq 1$ , with boundary conditions

$$E - v(0, t) - Ri(0, t) = 0, \quad (1.7)$$

$$C_1 \frac{dv(1, t)}{dt} = i(1, t) - g(v(1, t)). \quad (1.8)$$

The general solution of (1.5) and (1.6) is

$$v(x, t) = \phi(x - t/\tau) + \psi(x + t/\tau), \quad (1.9)$$

$$i(x, t) = k[\phi(x - t/\tau) - \psi(x + t/\tau)], \quad (1.10)$$

where  $\tau = \sqrt{LC}$  and  $k = \sqrt{C/L}$ . Let  $x = 1$  in (1.5) and (1.6) and solve for  $\phi$  and  $\psi$ , with a scaling and shift of time variable, the authors obtain

$$\phi(\alpha) = \frac{1}{2} \left[ p((1 - \alpha)\tau) + \frac{1}{k} q((1 - \alpha)\tau) \right], \quad (1.11)$$

$$\psi(\beta) = \frac{1}{2} \left[ p((\beta - 1)\tau) - \frac{1}{k} q((\beta - 1)\tau) \right], \quad (1.12)$$

where

$$p(t) = v(1, t), \quad (1.13)$$

$$q(t) = i(1, t). \quad (1.14)$$

After using (1.11) and (1.12) in (1.9) and (1.10) for  $x = 0$ , one arrives at

$$v(0, t) = \frac{1}{2} \left[ p(t + \tau) + p(t - \tau) + \frac{1}{k} q(t + \tau) - \frac{1}{k} q(t - \tau) \right], \quad (1.15)$$

$$i(0, t) = \frac{k}{2} \left[ p(t + \tau) - p(t - \tau) + \frac{1}{k} q(t + \tau) + \frac{1}{k} q(t - \tau) \right]. \quad (1.16)$$

By using (1.15), (1.16), (1.13), and (1.14) in the boundary conditions (1.7) and (1.8), with a shift of time, one obtains

$$\begin{aligned} E &= K_1 p(t) + K_2 p(t - 2\tau) + \frac{K_1}{k} q(t) - \frac{K_2}{k} q(t - 2\tau), \\ \dot{p}(t) &= \frac{1}{C_1} q(t) - \frac{1}{C_1} g(p(t)), \end{aligned}$$

where  $K_1 = (1 + kR)/2$  and  $K_2 = (1 - kR)/2$ . This is in the form of (1.3) and (1.4). A transformation of variables  $(p(t), q(t)) \rightarrow (p(t), r(t))$  with  $r(t) = p(t) - q(t)/k$  allows us transform the above to

$$\begin{aligned} \dot{p}(t) &= \left[ \frac{k}{C_1} p(t) - \frac{1}{C_1} g(p(t)) \right] - \frac{k}{C_1} r(t), \\ r(t) &= 2p(t) + \frac{K_2}{K_1} r(t - 2\tau) - \frac{E}{K_1}, \end{aligned}$$

which is in the standard form of (1.1) and (1.2).

Most early studies transform coupled DDEs to a set of regular DDEs of neutral type [21, 35]. A notable exception is [44]. In recent few years, there is a renewed interest in coupled DDEs. A substantial effort has been devoted to the Lyapunov–Krasovskii functional approach [9, 27, 28, 41–43, 45]. The reader is also referred to the plenary lecture by Rasvan [46] for a comprehensive review. Most studies are carried out on the seemingly more general form of (1.3) and (1.4). On the other hand, it should be pointed out that the description (1.1) and (1.2) is not less general than (1.3) and (1.4). Indeed, by introducing an additional variable  $z(t) = x(t)$ , (1.3) and (1.4) can be written as

$$\begin{aligned} \dot{x}(t) &= f(t, x(t), z_d(t), y_d(t)), \\ y(t) &= g(t, x(t), z_d(t), y_d(t)), \\ z(t) &= x(t), \end{aligned}$$

which is obviously in the form of (1.1) and (1.2).

It was suggested in [15, 16] that it is also of interest to reformulate regular DDEs, even of retarded type, in the form of coupled DDEs (1.1) and (1.2). Indeed, many practical systems may have a large number of state variables, but only involve a rather small number of delayed variables. A process similar to “pulling out uncertainties” as described in [6, 7] allows the authors to “pull out delays” and write the system in a form with all delays appearing in the feedback channel. In [15–17], the case of single delay or commensurate delays was described by the forward system from  $u$  to  $y$ ,

$$\begin{aligned} \dot{x}(t) &= \hat{f}(t, x(t), u(t)), \\ y(t) &= \hat{g}(t, x(t), u(t)), \end{aligned}$$

and feedback consisting of the delay

$$u(t) = y(t - r).$$

It should be pointed out that the “feedback” does not have to be the active action of any controller. Rather, it is often the nature of the component. Having the delay in the feedback path is a choice of modeling. In this chapter, the more general case of multiple delays is considered:

$$u(t) = h(y(t - r_1), y(t - r_2), \dots, y(t - r_K)).$$

This results in a standard form of coupled DDEs (1.1) and (1.2) where the dimension of  $x$  may be significantly larger than that of  $y$ . As will be shown later on, at least in the case of linear systems, the stability analysis of the system in this form using Lyapunov-Krasovskii functional approach may require much less computation than if it is formulated in a regular DDE form of Hale’s type

$$\begin{aligned} & \frac{d}{dt} \bar{g}(t, x(t), x(t - r_1), x(t - r_2), \dots, x(t - r_K)) \\ &= \bar{f}(t, x(t), x(t - r_1), x(t - r_2), \dots, x(t - r_K)). \end{aligned} \quad (1.17)$$

Not surprisingly, the known stability results of coupled DDEs (1.1) and (1.2) (or its more general form of coupled differential-functional equations) parallel those of regular DDEs of neutral type (1.17) (or its more general form of functional differential equations) discussed in [22, 31, 32]. Its flexibility in modeling practical systems makes it an ideal form in many practical applications, and warrants a substantial effort in developing further results relevant to such systems. In this chapter, some important stability results of coupled DDEs (1.1) and (1.2) and its generalization and specializations to linear systems are covered. This chapter and the next four chapters cover material pertaining to control and stability analysis of DDEs.

This first chapter is organized as follows. Notation is described in the remaining part of this section. In Sect. 2, the general stability theory of coupled differential-functional equations by using the Lyapunov-Krasovskii functional approach is discussed. In Sect. 3, the stability problem of continuous time difference equations is presented. Following that, in Sect. 4, the quadratic Lyapunov-Krasovskii functional for linear systems and the discretization process is detailed. Examples are included in Sect. 5 and discussed. Finally, concluding remarks are presented in Sect. 6.

Here,  $\mathbb{R}$  is the set of real numbers,  $\mathbb{R}^n$  and  $\mathbb{R}^{m \times n}$  represent the set of real  $n$ -vectors and  $m$  by  $n$  matrices, respectively.  $\mathbb{R}_+$  and  $\bar{\mathbb{R}}_+$  denote the sets of positive and nonnegative real numbers, respectively. For a given  $r \in \mathbb{R}_+$ , which usually corresponds to the delay,  $\mathcal{PC}$  is the set of bounded, right continuous, and piecewise continuous  $\mathbb{R}^n$ -valued functions defined on  $[-r, 0)$ . For a given bounded, right continuous, and piecewise continuous function  $y : [\sigma - r, \infty) \rightarrow \mathbb{R}^n$  and  $\tau \geq \sigma$ , the authors define  $y_\tau \in \mathcal{PC}$  by  $y_\tau(\theta) = y(\tau + \theta)$ ,  $\theta \in [-r, 0)$ .  $\|\cdot\|$  denotes 2-norm for vectors and the corresponding induced norm for matrices. For  $\phi \in \mathcal{PC}$ ,  $\psi \in \mathbb{R}^m$ , define  $\|\phi\| \triangleq \sup_{-r \leq \theta < 0} \|\phi(\theta)\|$  and  $\|(\psi, \phi)\| \triangleq \max\{\|\phi\|, \|\psi\|\}$ . For

$x : [\sigma, \infty) \rightarrow \mathbb{R}^m$  and  $\sigma \leq s < \tau$ ,  $x_{[s, \tau)}$  denotes the restriction of  $x$  in  $[s, \tau)$ , and  $\|x_{[s, \tau)}\| = \sup_{s \leq t < \tau} \|x(t)\|$ . For an  $A \in \mathbb{R}^{m \times n}$ ,  $A^T$  denotes the transpose of  $A$ . If  $A$  is square, then  $\rho(A)$  represents the greatest absolute value of all the eigenvalues of  $A$ . If  $A = A^T$ , the authors use  $A > 0$  and  $A \geq 0$  to denote  $A$  to be a positive definite and positive semidefinite matrix. Similar notation is used to denote negative definiteness and semidefiniteness.

## 1.2 Coupled Differential-Functional Equations

In this section, the authors present the basic stability theory of coupled differential-functional equations

$$\dot{x}(t) = f(t, x(t), y_t), \quad (1.18)$$

$$y(t) = g(t, x(t), y_t), \quad (1.19)$$

where  $x(t) \in \mathbb{R}^m$ ,  $y(t) \in \mathbb{R}^n$ . The notation  $y_t \in \mathcal{PC}$  represents a section of function  $y(t)$  from  $[t - r, t)$ , with the notation

$$y_t(\theta) = y(t + \theta), \quad \theta \in [-r, 0),$$

and  $r \in \mathbb{R}_+$  is the maximum delay. Obviously, the system (1.18) and (1.19) includes (1.1) and (1.2) as a special case. It is also interesting to point out that (1.18) and (1.19) and (1.1) and (1.2) may also be used to represent many singular systems.

The smallest possible initial time  $\sigma$  is given. The initial conditions may be defined for any  $t_0 \geq \sigma$  as

$$x(t_0) = \psi,$$

$$y_{t_0} = \phi,$$

where  $\psi \in \mathbb{R}^m$  and  $\phi \in \mathcal{PC}$ . As is conventional in stability study, it is assumed that the real functions  $f$  and  $g$  satisfy  $f(t, 0, 0) = g(t, 0, 0) = 0$ , and the stability of the trivial solution is studied. The material in this section is mainly adapted from [16].

The setting (1.18) and (1.19) is very general. If  $f$  and  $g$  are linear, then according to Riesz representation theorem, it is always possible to write

$$f(t, x(t), y_t) = A(t)x(t) + \int_{-r}^0 d_\theta \mu(t, \theta)y(t + \theta), \quad (1.20)$$

$$g(t, x(t), y_t) = C(t)x(t) + \int_{-r}^0 d_\theta \eta(t, \theta)y(t + \theta), \quad (1.21)$$

where the subscript  $\theta$  indicates that  $\theta$  is the integration variable, and the matrix functions  $\mu$  and  $\eta$  are of bounded variation with respect to  $\theta$  for any fixed  $t$ , and the integration is Riemann–Stieltjes integral. In practice, it is often sufficient to restrict to linear systems in the form of

$$f(t, x(t), y_t) = A(t)x(t) + \sum_{i=1}^K B_i(t)y(t-r_i) + \int_{-r}^0 B(t, \theta)y(t+\theta)d\theta, \quad (1.22)$$

$$g(t, x(t), y_t) = C(t)x(t) + \sum_{i=1}^K D_i(t)y(t-r_i) + \int_{-r}^0 D(t, \theta)y(t+\theta)d\theta, \quad (1.23)$$

where  $0 < r_i \leq r$ ,  $i = 1, 2, \dots, K$ , and

$$\int_{-r}^0 \|B(t, \theta)\|d\theta < \infty, \quad \int_{-r}^0 \|D(t, \theta)\|d\theta < \infty.$$

As bounded variation implies at most countable discontinuities, the system expressed by (1.20) and (1.21) would be equivalent to (1.22 and 1.23) if the number of discrete delays  $K$  is allowed to be  $\infty$  with the restriction

$$\sum_{i=1}^{\infty} \|B_i(t)\| < \infty \quad \text{and} \quad \sum_{i=1}^{\infty} \|D_i(t)\| < \infty.$$

It is assumed that there exists a unique solution  $(x(t), y_t)$  to the equations in  $[t_0, +\infty)$  for any initial conditions  $t_0 \geq \sigma$ ,  $\psi \in \mathbb{R}^m$ , and  $\phi \in \mathcal{PC}$ . Such solutions are often denoted as  $x(t; t_0, \psi, \phi)$  and  $y_t(t_0, \psi, \phi)$  when the explicit dependence on the initial conditions is important.

Although it is not the purpose of this chapter to study the existence and uniqueness of the solution, it can be shown using a similar procedure to [20] that the equations have a unique solution if  $f(t, \psi, \phi)$  and  $g(t, \psi, \phi)$  are continuous with respect to  $t$  and Lipschitz with respect to  $\psi$  and  $\phi$ , and the function  $g$  is *uniformly nonatomic at zero*. The function  $g$  is uniformly non-atomic at zero if there exists a  $\mu > 0$  and  $s_0 > 0$  and a strictly increasing function  $\zeta(\cdot)$ ,  $0 \leq \zeta(s) < 1$  for all  $s \in [0, s_0)$ , such that

$$\|g(t, x, \phi) - g(t, x, \tilde{\phi})\| \leq \zeta(s)\|\phi - \tilde{\phi}\|,$$

for all  $\phi, \tilde{\phi} \in \mathcal{PC}$ ,  $\|\phi - \tilde{\phi}\| \leq \mu$ ,  $\tilde{\phi}(\theta) = \phi(\theta)$  for  $\theta \in [-r, -s]$ .

In this chapter, the state  $(x(t), y_t)$  evolves in  $\mathbb{R}^m \times \mathcal{PC}$ . Let  $\mathcal{C}$  be the continuous bounded  $\mathbb{R}^n$ -valued functions defined in  $[-r, 0]$ . In the literature, it is not uncommon to constrain the state to evolve within the set

$$\{(\psi, \phi) \in \mathbb{R}^m \times \mathcal{C} \mid \lim_{\theta \rightarrow 0} \phi(\theta) = \lim_{\theta \rightarrow 0} g(t, \psi, \phi(\theta))\}.$$

Indeed, if the initial condition satisfies the above constraint, then the state also satisfies the above constraint [21]. Almost all the conclusions in this chapter apply to this case.

Another possibility is to relax the state to evolve in  $\mathbb{R}^m \times L_2$  such as was done in [41, 42]. The theory discussed in this chapter may not be directly applicable to this possibility mainly due to different norms used.

The definition of stability is very similar to the normal time-delay systems, and is presented in the following.

**Definition 1.1.** The trivial solution  $x(t) = y(t) = 0$  is said to be stable if for any  $t_0 \geq \sigma$  and any  $\varepsilon > 0$ , there exists a  $\delta = \delta(t_0, \varepsilon) > 0$  such that  $\|(x(t_0), y_{t_0})\| < \delta$  implies  $\|(x(t), y_t)\| < \varepsilon$  for all  $t > t_0$ . It is said to be asymptotically stable if it is stable, and for any  $t_0 \in \mathbb{R}$  there exists a  $\delta_a = \delta_a(t_0) > 0$  such that  $\|(x(t_0), y_{t_0})\| < \delta_a$  implies  $x(t) \rightarrow 0, y(t) \rightarrow 0$  as  $t \rightarrow \infty$ . It is said to be uniformly stable if it is stable and  $\delta(t_0, \varepsilon)$  can be chosen to be independent of  $t_0$ . It is said to be uniformly asymptotically stable if it is uniformly stable, and there exists a  $\delta_a > 0$  independent of  $t_0$  such that for any  $\eta > 0$ , there exists a  $T = T(\delta_a, \eta)$  such that  $\|(x(t_0), y_{t_0})\| < \delta_a$  implies  $\|(x(t), y_t)\| < \eta$  for all  $t \geq t_0 + T$  and  $t_0 \geq \sigma$ . It is globally (uniformly) asymptotically stable if it is (uniformly) asymptotically stable and  $\delta_a$  can be an arbitrarily large finite number.

An important component of studying the stability of the complete system consisting of (1.18) and (1.19) is understanding the subsystem (1.19). Taken alone, (1.19) can be considered as a system described by the function  $g$  with  $x$  as the input and  $y_t$  as the state. From this point of view,  $y_t$  depends on the initial condition  $y_{t_0} = \phi$  and the input  $x_{[t_0, t]}$ , and can be denoted as  $y_t(t_0, \phi, x)$ . The following definition is introduced.

**Definition 1.2.** The function  $g$  or the subsystem (1.19) defined by  $g$  is said to be *input-to-state stable* if for any  $t_0$ , there exist a  $\mathcal{KL}$  function  $\beta$  (i.e.,  $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$ ,  $\beta(\alpha, t)$  is continuous, strictly increasing with respect to  $\alpha$ , strictly decreasing with respect to  $t$ ,  $\beta(0, t) = 0$ , and  $\lim_{t \rightarrow \infty} \beta(\alpha, t) = 0$ ) and a  $\mathcal{K}$  function  $\gamma$  (i.e.,  $\gamma : \mathbb{R}_+ \rightarrow \mathbb{R}_+$  is continuous, strictly increasing, and  $\gamma(0) = 0$ ), such that the solution  $y_t(t_0, \phi, x)$  corresponding to the initial condition  $y_{t_0} = \phi$  and input  $x(t)$  satisfies

$$\|y_t(t_0, \phi, x)\| \leq \beta(\|\phi\|, t - t_0) + \gamma(\|x_{[t_0, t]}\|). \quad (1.24)$$

If  $\beta$  and  $\gamma$  can be chosen to be independent of  $t_0$ , then it is said to be *uniformly input-to-state stable*.

This definition is along the line of literature on input-to-state stability for continuous time systems [49, 50], discrete time systems [25], and time-delay systems [52]. A well-known example of input-to-state stable system is

$$g(t, x(t), y_t) = Cx(t) + Dy(t - r),$$

with  $D$  satisfying  $\rho(D) < 1$  [25]. In this case [22], for any  $0 < \delta < \frac{-\ln \rho(D)}{r}$ , there exists an  $M > 0$  such that

$$\|y_t(t_0, x, \phi)\| \leq M\|y_{t_0}\|e^{-\delta(t-t_0)} + \sup_{t_0 \leq \tau < t} \|x(\tau)\|,$$

and the input-to-state stability is established by letting

$$\begin{aligned}\beta(\alpha, t) &= M\alpha e^{-\delta t}, \\ \gamma(\alpha) &= M\alpha.\end{aligned}$$

With the above background the Lyapunov-Krasovskii functional method is now presented to study the stability of systems described by (1.18) and (1.19). Let  $V(t, \psi, \phi)$  be differentiable, and define

$$\begin{aligned}\dot{V}(\tau, \psi, \phi) &\triangleq \frac{d}{dt}V(t, x(t), y_t) \Big|_{t=\tau, x(\tau)=\psi, y_\tau=\phi} \\ &= \limsup_{t \rightarrow \tau+} \frac{V(t, x(t; \tau, \psi, \phi), y_t(\tau, \psi, \phi)) - V(\tau, \psi, \phi)}{t - \tau}.\end{aligned}$$

Then the following theorem can be presented.

**Theorem 1.1.** *Suppose that  $f$  and  $g$  maps  $\mathbb{R} \times (\text{bounded sets in } \mathbb{R}^m \times \mathcal{PC})$  into bounded sets in  $\mathbb{R}^m$  and  $\mathbb{R}^n$ , respectively, and  $g$  is uniformly input-to-state stable;  $u, v, w : \bar{\mathbb{R}}_+ \rightarrow \bar{\mathbb{R}}_+$  are continuous nondecreasing functions, where additionally  $u(s)$  and  $v(s)$  are positive for  $s > 0$ , and  $u(0) = v(0) = 0$ . If there exists a continuous differentiable functional  $V : \mathbb{R} \times \mathbb{R}^n \times \mathcal{PC} \rightarrow \mathbb{R}$  such that*

$$u(\|\psi\|) \leq V(t, \psi, \phi) \leq v(\|(\psi, \phi)\|), \quad (1.25)$$

and

$$\dot{V}(t, \psi, \phi) \leq -w(\|\psi\|), \quad (1.26)$$

then the trivial solution of the coupled differential-functional equations (1.18) and (1.19) is uniformly stable. If  $w(s) > 0$  for  $s > 0$ , then it is uniformly asymptotically stable. If, in addition,  $\lim_{s \rightarrow \infty} u(s) = \infty$ , then it is globally uniformly asymptotically stable.

*Proof.* To prove uniform stability, for any given  $\varepsilon > 0$ , one may find a corresponding  $\delta = \delta(\varepsilon)$  as follows: Choose  $\hat{\varepsilon} > 0$ ,  $\hat{\varepsilon} < \varepsilon$ , such that  $\beta(\hat{\varepsilon}, 0) < \varepsilon/2$ ,  $\gamma(\hat{\varepsilon}) < \varepsilon/2$ , and let  $\delta > 0$  be such that  $\delta \leq \hat{\varepsilon}$  and  $v(\delta) < u(\hat{\varepsilon})$ . Then, for any  $\|(\psi, \phi)\| < \delta$ , due to (1.25) and (1.26), one has

$$\begin{aligned}u(\|x(t)\|) &\leq V(t, x(t), y_t) \leq V(t_0, \psi, \phi) \\ &\leq v(\|(\psi, \phi)\|) \leq v(\delta) < u(\hat{\varepsilon}),\end{aligned}$$

which implies  $\|x(t)\| < \hat{\varepsilon} < \varepsilon$ . Furthermore,

$$\begin{aligned}\|y_t(t_0, \phi, x)\| &\leq \beta(\|\phi\|, t) + \gamma(\|x_{[t_0, t]}\|) \\ &< \varepsilon/2 + \varepsilon/2 \\ &= \varepsilon.\end{aligned}$$

Thus  $\|(x(t), y_t)\| < \varepsilon$  and uniform stability is proven.

Now let  $\varepsilon_a = 1$ . Because of uniform stability, there exists a  $\delta_a = \delta(\varepsilon_a)$  such that  $\|(x(t), y_t)\| < \varepsilon_a$  for any  $t_0 \geq \sigma$  and  $\|(\psi, \phi)\| < \delta_a$ . To show uniform asymptotic

stability, one needs to show further that for any  $\eta > 0$ , one can find a  $T = T(\delta_a, \eta)$  such that  $\|(x(t), y_t)\| < \eta$  for all  $t \geq t_0 + T$  and  $\|(\psi, \phi)\| < \delta_a$ . Because of uniform stability, let  $\delta = \delta(\eta)$  as was done in proving uniform stability, it is sufficient to show that there exists a  $t \in (t_0, t_0 + T]$  such that  $\|(x(t), y_t)\| < \delta$ .

Let  $T_a$  be such that

$$\beta(\epsilon_a, T_a) \leq \delta/2, \quad (1.27)$$

and

$$\alpha = \min\{\gamma^{-1}(\delta/2), \delta\}. \quad (1.28)$$

As  $f$  maps bounded sets to bounded sets, there exists an  $L > 0$  such that

$$\|f(t, x(t), y_t)\| < L \quad \text{for all } t \geq t_0 \text{ and } \|(x(t), y_t)\| \leq \epsilon_a. \quad (1.29)$$

One may increase  $L$  if necessary to satisfy

$$\alpha/L < T_a.$$

Then,  $T$  may be chosen as

$$T = (2K + 1)T_a,$$

where  $K$  is the smallest integer to satisfy  $K > \frac{Lv(\epsilon_a)}{\alpha w(\alpha/2)}$ . If the statement is not true; that is,

$$\|(x(t), y_t)\| \geq \delta, \quad (1.30)$$

for all  $t \in (t_0, t_0 + T]$ . Then, it can be shown that there exists a  $s_k \in [t_k - T_a, t_k]$ ,  $t_k = t_0 + 2kT_a$ ,  $k = 1, 2, \dots, K$ , such that

$$\|x(s_k)\| \geq \alpha. \quad (1.31)$$

Indeed, (1.30) implies either

$$\|x(t_k)\| \geq \delta,$$

in which case (1.31) is satisfied, or for  $s = t_k - T_a$

$$\begin{aligned} \delta &\leq \|y_{t_k}\| \\ &\leq \beta(\|y_s\|, t - s) + \gamma(\|x_{[s, t_k]}\|) \\ &\leq \beta(\epsilon_a, T_a) + \gamma(\|x_{[s, t_k]}\|) \\ &\leq \delta/2 + \gamma(\|x_{[s, t_k]}\|), \end{aligned}$$

which implies  $\gamma(\|x_{[s, t_k]}\|) \geq \delta/2$ , or  $\|x_{[s, t_k]}\| \geq \alpha$ , i.e., there is a  $s_k \in [t_k - T_a, t_k]$  such that (1.31) is satisfied.

From (1.31), (1.29), and (1.18), it can be concluded that, for  $t \in I_k = [s_k - \frac{\alpha}{2L}, s_k + \frac{\alpha}{2L}]$ ,

$$\|x(t)\| \geq \|x(s_k)\| - L \cdot \frac{\alpha}{2L} \geq \alpha/2.$$

This implies that  $\dot{V}(t, x(t), y_t) \leq -w(\alpha/2)$  in  $t \in I_k, k = 1, 2, \dots, K$ , and  $\dot{V}(t, x(t), y_t) \leq 0$  elsewhere, and  $I_k$  have no overlap for distinct  $k$ . This implies that for  $t = t_0 + T$ ,

$$\begin{aligned} V(t, x(t), y_t) &\leq V(t_0, \psi, \phi) - Kw(\alpha/2)\alpha/L \\ &\leq v(\varepsilon_a) - K\alpha w(\alpha/2)/L \\ &< 0, \end{aligned}$$

which contradicts the fact that  $V(t, x(t), y_t) \geq 0$ . This proved asymptotic stability.

Finally, if  $\lim_{s \rightarrow \infty} u(s) = \infty$ , then  $\delta_a$  above may be arbitrarily large, and  $\varepsilon_a$  can be chosen after  $\delta_a$  is given to satisfy  $v(\delta_a) < u(\varepsilon_a)$ , and therefore global asymptotic stability can be concluded.

This theorem is due to Gu and Liu [16], and is very similar to the counterpart of functional differential equations of neutral type shown in [5]. In [15], asymptotic, but not uniform, stability was proved under the assumption that  $g$  is input-to-state stable. Asymptotic stability was also established in [43] for systems with multiple discrete delays.

The rest of this first chapter is devoted to linear systems. For linear systems, uniform stability is equivalent to exponential stability.

### 1.3 Stability of Continuous Time Difference Equations

This section considers the stability problem of a system described by

$$y(t) = \sum_{i=1}^K D_i y(t - r_i) + h(t), \quad (1.32)$$

and the corresponding homogeneous equation

$$y(t) = \sum_{i=1}^K D_i y(t - r_i), \quad (1.33)$$

where  $y(t) \in \mathbb{R}^n$  and  $D_i \in \mathbb{R}^{n \times n}$ . Let

$$r = \max_{1 \leq i \leq K} r_i.$$

Then, initial condition can be expressed as

$$y_{t_0} = \phi,$$

where  $y_t$  is defined by

$$y_t(\theta) = y(t + \theta), \quad -r \leq \theta < 0.$$

The characteristic equation of the system is

$$\Delta(\lambda) = \det\left[I - \sum_{i=1}^K e^{-r_i \lambda} D_i\right] = 0. \quad (1.34)$$

It is well known [22] that

$$a_D = \sup\{\operatorname{Re}(\lambda) \mid \Delta(\lambda) = 0\} < \infty. \quad (1.35)$$

For any  $\delta > a_D$ , there exists an  $M > 0$  such that the solution to (1.32) satisfies

$$\|y_t(t_0, \phi, h)\| \leq M[\|\phi\|e^{\delta(t-t_0)} + \sup_{t_0 \leq \tau \leq t} \|h(\tau)\|].$$

See Theorem 3.4 and the discussion before the Theorem in Chap. 9 of [22]. A consequence of this is that (1.33) is exponentially stable and (1.32) is uniformly input-to-state stable if and only if  $a_D < 0$ .

Unlike time-delay systems of retarded type, the stability of difference equations may change drastically under small deviation of delay ratio. First, the following concepts are introduced.

**Definition 1.3.** A set of positive real scalars  $r_1, r_2, \dots, r_K$  are rationally independent if the equation

$$\sum_{i=1}^K \alpha_i r_i = 0,$$

can be satisfied for rational numbers  $\alpha_i$ ,  $i = 1, 2, \dots, K$  only if  $\alpha_i = 0$  for all  $i$ .

Obviously,  $\alpha_i$  can be replaced by integers without loss of generality.

**Theorem 1.2.** With given coefficient matrices  $D_i$ ,  $i = 1, 2, \dots, K$ , and “stable” meaning  $a_D < 0$ , the following statements are equivalent about the system (1.33):

- (i) The system is stable for a set of rationally independent delays  $r_i$ ,  $i = 1, 2, \dots, K$ ;
- (ii) For given nominal delays  $r_i^0 > 0$ ,  $i = 1, 2, \dots, K$  and arbitrarily small fixed deviation bound  $\varepsilon > 0$ . The system is stable for any set of delays  $r_i$ ,  $i = 1, 2, \dots, K$  that satisfy

$$|r_i - r_i^0| < \varepsilon;$$

- (iii) The system is stable for all

$$r_i > 0, \quad i = 1, 2, \dots, K;$$

- (iv) The coefficient matrices satisfy

$$\sup \left\{ \rho \left( \sum_{i=1}^K e^{j\theta_i} A_i \right) \mid \theta_i \in [0, 2\pi], i = 1, 2, \dots, K \right\} < 1,$$

where  $j$  is the imaginary unit.

The above theorem is due to Hale [19], and can be found in [48] and as Theorem 6.1 of Chap. 9 in [22]. The above theorem indicates that if a difference equation is exponentially stable under an arbitrarily small independent deviation of delays, then it must also be stable for arbitrary delays.

On the other hand, there are indeed practical cases where rational dependence of delays are not subject to error due to the structure of the problem. In these cases, the system can usually be recast in a different form, so that only rationally independent parameters appear as delays as will be illustrated later in this chapter.

The following theorem presents the stability condition in a Lyapunov-like form.

**Theorem 1.3.** *If there exist symmetric positive definite matrices  $S_1, S_2, \dots, S_K$  such that*

$$\begin{pmatrix} D_1^T \\ D_2^T \\ \vdots \\ D_K^T \end{pmatrix} \sum_{i=1}^K S_i (D_1 \ D_2 \ \dots \ D_K) - \text{diag} (S_1 \ S_2 \ \dots \ S_K) < 0 \quad (1.36)$$

*is satisfied, then  $a_D < 0$  for the system (1.33).*

It is easy to see that the above theorem is equivalent to Theorem 6.1 of [3]. The proof there uses (iv) of Theorem 1.2. In the following, a more direct proof is provided.

*Proof.* If (1.36) is true, then

$$\begin{pmatrix} D_1^T \\ D_2^T \\ \vdots \\ D_K^T \end{pmatrix} \sum_{i=1}^K S_i (D_1 \ D_2 \ \dots \ D_K) - \text{diag} (S_1 \ S_2 \ \dots \ S_K) \leq -\varepsilon \text{diag} (S_1 \ S_2 \ \dots \ S_K) \quad (1.37)$$

for some sufficiently small  $\varepsilon > 0$ . Let  $\lambda$  be a solution to the characteristic equation (1.34). Then there exists a  $\zeta \neq 0$  such that

$$[I - \sum_{i=1}^K e^{-r_i \lambda} D_i] \zeta = 0,$$

or

$$\sum_{i=1}^K e^{-r_i \lambda} D_i \zeta = \zeta. \quad (1.38)$$

Right multiply (1.37) by

$$\begin{pmatrix} e^{-r_1 \lambda} \zeta \\ e^{-r_2 \lambda} \zeta \\ \vdots \\ e^{-r_K \lambda} \zeta \end{pmatrix},$$

and left multiply by its conjugate transpose; one obtains

$$\begin{aligned} & \left[ \sum_{i=1}^K e^{-r_i \lambda} D_i \zeta \right]^* \left[ \sum_{i=1}^K S_i \right] \left[ \sum_{i=1}^K e^{-r_i \lambda} D_i \zeta \right] - \sum_{i=1}^K e^{-2r_i \sigma} \zeta^* S_i \zeta \\ & \leq -\varepsilon \sum_{i=1}^K e^{-2r_i \sigma} \zeta^* S_i \zeta. \end{aligned}$$

After using (1.38), and letting  $\sigma = \operatorname{Re}(\lambda)$ , the above becomes

$$\zeta^* \left[ \sum_{i=1}^K S_i \right] \zeta - (1 - \varepsilon) \sum_{i=1}^K e^{-2r_i \sigma} \zeta^* S_i \zeta \leq 0,$$

or

$$\zeta^* \left[ \sum_{i=1}^K (1 - (1 - \varepsilon)e^{-2r_i \sigma}) S_i \right] \zeta \leq 0.$$

As  $\zeta \neq 0$  and  $S_i > 0$  for all  $i$ , the above can be satisfied only if

$$1 - (1 - \varepsilon)e^{-2r_i \sigma} < 0 \text{ for some } i,$$

or

$$\sigma \leq -\min_{1 \leq r_i \leq K} \frac{\ln(\frac{1}{1-\varepsilon})}{2r_i} \triangleq -\delta.$$

As the above must be satisfied for all solutions of the characteristic equation (1.34), it can be concluded that  $a_D \leq -\delta$ .

## 1.4 Linear Coupled Differential-Difference Equations

### 1.4.1 Quadratic Lyapunov-Krasovskii Functional

In this section, the system described by the following coupled linear differential-difference equations is discussed

$$\dot{x}(t) = Ax(t) + \sum_{i=1}^K B_i y(t - r_i), \quad (1.39)$$

$$y(t) = Cx(t) + \sum_{i=1}^K D_i y(t - r_i), \quad (1.40)$$

where  $x(t) \in \mathbb{R}^m$  and  $y(t) \in \mathbb{R}^n$  are the state variables;  $A \in \mathbb{R}^{m \times m}$ ,  $C \in \mathbb{R}^{n \times m}$ ,  $B_i \in \mathbb{R}^{m \times n}$ ,  $D_i \in \mathbb{R}^{m \times m}$ ,  $i = 1, 2, \dots, K$  are the coefficient matrices. The initial conditions are defined as

$$\begin{aligned} x(t_0) &= \psi, \\ y_{t_0} &= \phi. \end{aligned}$$

Without loss of generality, the delays are ordered such that

$$0 < r_1 < r_2 < \cdots < r_K = r.$$

Similar to [12] and Sect. 7.5 of [14], choose Lyapunov-Krasovskii functional in the form of

$$\begin{aligned} V(\psi, \phi) &= \psi^T P \psi + 2\psi^T \sum_{i=1}^K \int_{-r_i}^0 Q^i(\eta) \phi(\eta) d\eta \\ &\quad + \sum_{i=1}^K \sum_{j=1}^K \int_{-r_i}^0 \int_{-r_j}^0 \phi^T(\xi) R^{ij}(\xi, \eta) \phi(\eta) d\xi d\eta \\ &\quad + \sum_{i=1}^K \int_{-r_i}^0 \phi^T(\eta) S^i(\eta) \phi(\eta) d\eta, \end{aligned} \quad (1.41)$$

where

$$\begin{aligned} P &= P^T \in \mathbb{R}^{m \times m}, \\ Q^i(\eta) &\in \mathbb{R}^{m \times n}, \\ R^{ij}(\xi, \eta) &= R^{jiT}(\eta, \xi) \in \mathbb{R}^{n \times n}, \\ S^i(\eta) &= S^{iT}(\eta) \in \mathbb{R}^{n \times n}, \end{aligned} \quad (1.42)$$

for  $i, j = 1, 2, \dots, K$ ;  $\xi, \eta \in [-r, 0]$ . The functions  $Q^i(\xi)$ ,  $S^i(\xi)$ ,  $R^{ij}(\xi, \eta)$ , and  $R^{iK}(\xi, \eta)$ ,  $i, j = 1, 2, \dots, K-1$  are introduced to account for discontinuities, and therefore, it is sufficient to constrain these functions to the following special forms without loss of generality:

$$\begin{aligned} Q^i(\xi) &= Q^i = \text{constant}, \\ S^i(\xi) &= S^i = \text{constant}, \\ R^{ij}(\xi, \eta) &= R^{ij} = \text{constant}, \\ R^{iK}(\xi, \eta) &= R^{iK}(\eta) = R^{KiT}(\eta) \text{ independent of } \xi, \end{aligned} \quad (1.43)$$

for  $i, j = 1, 2, \dots, K-1$ . It should be pointed out that no such constraint is applied to  $R^{KK}$ .

The derivative of  $V(\psi, \phi)$  in (1.41) along the system trajectory can be calculated as

$$\begin{aligned} \dot{V}(t, \psi, \phi) &= - \sum_{i=0}^K \sum_{j=0}^K \varphi^T(-r_i) \bar{\Delta}_{ij} \varphi(-r_j) \\ &\quad + 2 \sum_{i=0}^K \sum_{j=1}^K \varphi^T(-r_i) \int_{-r_j}^0 \Pi^{ij}(\eta) \phi(\eta) d\eta \end{aligned}$$

$$\begin{aligned}
& -2 \sum_{i=1}^K \int_{-r_i}^0 \phi^T(\xi) d\xi \int_{-r}^0 \dot{R}^{iK}(\eta) \phi(\eta) d\eta \\
& - \int_{-r}^0 \int_{-r}^0 \phi^T(\xi) \left[ \frac{\partial}{\partial \xi} R^{KK}(\xi, \eta) + \frac{\partial}{\partial \eta} R^{KK}(\xi, \eta) \right] \phi(\eta) d\eta d\xi \\
& - \int_{-r}^0 \phi^T(\eta) \dot{S}^K(\eta) \phi(\eta) d\eta,
\end{aligned} \tag{1.44}$$

where

$$\varphi(-r_i) = \begin{cases} \psi, & i = 0; \\ \phi(-r_i), & 1 \leq i \leq K; \end{cases}$$

$$\begin{aligned}
\bar{\Delta}_{00} = & -[A^T P + P A + \sum_{l=1}^{K-1} (Q^l C + C^T Q^{lT} + C^T S^l C) \\
& + Q^K(0)C + C^T Q^{KT}(0) + C^T S^K(0)C],
\end{aligned} \tag{1.45}$$

$$\begin{aligned}
\bar{\Delta}_{0j} = & -PB_j - \sum_{l=1}^{K-1} Q^l D_j + Q^j - Q^K(0)D_j \\
& - \sum_{l=1}^{K-1} C^T S^l D_j - C^T S^K(0)D_j,
\end{aligned}$$

$$\begin{aligned}
\bar{\Delta}_{0K} = & -PB_K - \sum_{l=1}^{K-1} Q^l D_K + Q^K(-r) - Q^K(0)D_K \\
& - \sum_{l=1}^{K-1} C^T S^l D_K - C^T S^K(0)D_K,
\end{aligned}$$

$$\begin{aligned}
\bar{\Delta}_{ij} = & -D_i^T \left[ \sum_{l=1}^{K-1} S^l + S^K(0) \right] D_j, \quad 1 \leq i, j \leq K, \quad i \neq j, \\
\bar{\Delta}_{ii} = & S^i - D_i^T \left[ \sum_{l=1}^{K-1} S^l + S^K(0) \right] D_i, \quad 1 \leq i \leq K-1, \\
\bar{\Delta}_{KK} = & S^K(-r) - D_K^T \left[ \sum_{l=1}^{K-1} S^l + S^K(0) \right] D_K;
\end{aligned} \tag{1.46}$$

and

$$\begin{aligned}
\Pi_{0j} = & A^T Q^j + \sum_{l=1}^{K-1} C^T R^{lj} + C^T R^{jKT}(0), \\
\Pi_{0K} = & A^T Q^K(\eta) + \sum_{l=1}^{K-1} C^T R^{lK}(\eta) + C^T R^{KK}(0, \eta) - \dot{Q}^K(\eta) \\
\Pi_{ij} = & B_i^T Q^j + D_i^T \sum_{l=1}^{K-1} R^{lj} - R^{ij} + D_i^T R^{jKT}(0),
\end{aligned}$$

$$\begin{aligned}
\Pi_{Kj} &= B_K^T Q^j + D_K^T \sum_{l=1}^{K-1} R^{lj} + D_K^T R^{jK^T}(0) - R^{jK^T}(-r), \\
\Pi_{iK} &= B_i^T Q^K(\eta) + D_i^T \sum_{l=1}^{K-1} R^{lK}(\eta) - R^{iK}(\eta) + D_i^T R^{KK}(0, \eta), \\
\Pi_{KK} &= B_K^T Q^K(\eta) + D_K^T \sum_{l=1}^{K-1} R^{lK}(\eta) + D_K^T R^{KK}(0, \eta) - R^{KK}(-r, \eta), \\
1 \leq i \leq K-1, \quad 1 \leq j \leq K-1.
\end{aligned}$$

According to Theorem 1.1, one can conclude

**Theorem 1.4.** *The system described by (1.39) and (1.40) is exponentially stable if (1.40) is uniformly input-to-state stable, and the Lyapunov-Krasovskii functional (1.41) satisfies*

$$\varepsilon \|\psi\|^2 \leq V(t, \psi, \phi), \quad (1.47)$$

and its derivative (1.44) satisfies

$$\dot{V}(t, \psi, \phi) \leq -\varepsilon \|\psi\|^2. \quad (1.48)$$

for some  $\varepsilon > 0$ .

*Proof.* Obviously, for the expression (1.41), there exists a sufficiently large  $M$  such that

$$V(t, \psi, \phi) \leq M \|(\psi, \phi)\|^2.$$

This together with (1.47), (1.48), and the uniform input-to-state stability of (1.40), implies the uniform asymptotic stability of the system according to Theorem 1.1. As this is a linear system, uniform asymptotic stability is equivalent to exponential stability.  $\square$

The above theorem reduces the stability problem to checking the satisfaction of (1.47) and (1.48), and the uniform input-to-state stability of (1.40). For the special case of  $D_i = 0$ , for all  $i \neq K$ , combining the ideas of [16] and Chap. 7 of [14], it can be shown that the existence of a quadratic Lyapunov-Krasovskii functional in the form of (1.41) to satisfy the conditions of Theorem 1.4 is also necessary for exponential stability. In the general case, such a theoretical result has not been established. However, as is shown in [13], it is of interest to transform the system to a more standard form such that  $y$  is partitioned  $(y_1, y_2, \dots, y_K)$ , and each  $y_i$  is associated with one delay. More definitive theoretical results are available for such standard form in [13].

## 1.4.2 Discretization

To render the conditions in Theorem 1.4 computable, the authors introduce a discretization similar to [12] by restricting the functions  $Q^i$ ,  $R^{ij}$ , and  $S^i$  to be piecewise

linear. Specially, divide the interval  $[-r, 0]$  into  $N$  segments compatible with the delays such that  $-r_i, i = 1, 2, \dots, K-1$  are among the dividing points. In other words, let  $\theta_p, p = 0, 1, \dots, N$  be the dividing points,

$$0 = \theta_0 > \theta_1 > \dots > \theta_N = -r,$$

then

$$-r_i = \theta_{N_i}, \quad i = 1, 2, \dots, K.$$

Thus, the interval  $[-r_i, 0]$  is divided into  $N_i$  smaller intervals. Let  $h_p$  be the length of the  $p$ th segment,

$$h_p = \theta_{p-1} - \theta_p.$$

For the sake of convenience, define

$$N_0 = 0, \quad h_0 = 0, \quad h_{N+1} = 0, \quad \theta_{N+1} = \theta_N = -r.$$

Then, one has

$$0 = N_0 < N_1 < \dots < N_K = N$$

and

$$r_i = \sum_{p=1}^{N_i} h_p, \quad i = 1, 2, \dots, K.$$

The functions  $Q^K(\xi), S^K(\xi), R^{iK}(\xi)$ , and  $R^{KK}(\xi, \eta)$  are chosen to be piecewise linear as follows:

$$Q^K(\theta_p + \alpha h_p) = (1 - \alpha)Q_p^K + \alpha Q_{p-1}^K, \quad (1.49)$$

$$S^K(\theta_p + \alpha h_p) = (1 - \alpha)S_p^K + \alpha S_{p-1}^K, \quad (1.50)$$

$$R^{iK}(\theta_p + \alpha h_p) = (1 - \alpha)R_p^{iK} + \alpha R_{p-1}^{iK} \quad (1.51)$$

for  $0 \leq \alpha \leq 1; p = 1, 2, \dots, N, i = 1, 2, \dots, K-1$ ; and

$$\begin{aligned} & R^{KK}(\theta_p + \alpha h_p, \theta_q + \beta h_q) \\ &= \begin{cases} (1 - \alpha)R_{pq}^{KK} + \beta R_{p-1, q-1}^{KK} + (\alpha - \beta)R_{p-1, q}^{KK}, & \text{for } \alpha \geq \beta; \\ (1 - \beta)R_{pq}^{KK} + \alpha R_{p-1, q-1}^{KK} + (\beta - \alpha)R_{p, q-1}^{KK}, & \text{for } \alpha < \beta; \end{cases} \end{aligned} \quad (1.52)$$

for  $0 \leq \alpha \leq 1, 0 \leq \beta \leq 1$ ; and  $p, q = 1, 2, \dots, N$ . Then,  $V(\psi, \phi)$  is completely determined by the matrices  $P, Q_p^i, Q_p^K, S_p^i, S_p^K, R^{ij}, R^{iK}$  and  $R_{pq}^{KK}$ ,  $i, j = 1, \dots, K-1$ ,  $p, q = 0, 1, \dots, N$ . The stability problem becomes one of determining the input-to-state stability of (1.40), and the existence of these matrices such that both (1.47) and (1.48) are satisfied. The following theorem establishes the conditions for (1.47).

**Theorem 1.5.** *The Lyapunov-Krasovskii functional  $V$  in (1.41), with  $Q^K, S^K, R^{iK}$  and  $R^{KK}$  piecewise linear as expressed in (1.49)–(1.52), satisfies (1.47) if*

$$S^i > 0, i = 0, 1, \dots, K-1, \quad (1.53)$$

$$S_p^K > 0, p = 0, 1, \dots, N, \quad (1.54)$$

and

$$\begin{pmatrix} P & \bar{Q} & \tilde{Q}^K \\ \bar{Q}^T & \bar{R} + \bar{S} & \hat{R}^K - \bar{S}F \\ \tilde{Q}^{KT} & (\hat{R}^K - \bar{S}F)^T & \hat{R}^{KK} + \tilde{S}' + F^T \bar{S}F \end{pmatrix} > 0, \quad (1.55)$$

where

$$\begin{aligned} \bar{Q} &= (Q^1 \ Q^2 \ \dots \ Q^{K-1}), \\ \tilde{Q}^K &= (Q_0^K \ Q_1^K \ \dots \ Q_N^K), \end{aligned}$$

$$\bar{R} = \begin{pmatrix} R^{11} & R^{12} & \dots & R^{1,K-1} \\ R^{21} & R^{22} & \dots & R^{2,K-1} \\ \vdots & \vdots & \ddots & \vdots \\ R^{K-1,1} & R^{K-1,2} & \dots & R^{K-1,K-1} \end{pmatrix},$$

$$\hat{R}^K = \begin{pmatrix} R_0^{1K} & R_1^{1K} & \dots & R_N^{1K} \\ R_0^{2K} & R_1^{2K} & \dots & R_N^{2K} \\ \vdots & \vdots & \ddots & \vdots \\ R_0^{K-1,K} & R_1^{K-1,K} & \dots & R_N^{K-1,K} \end{pmatrix},$$

$$\hat{R}^{KK} = \begin{pmatrix} R_{00}^{KK} & R_{01}^{KK} & \dots & R_{0N}^{KK} \\ R_{10}^{KK} & R_{11}^{KK} & \dots & R_{1N}^{KK} \\ \vdots & \vdots & \ddots & \vdots \\ R_{N0}^{KK} & R_{N1}^{KK} & \dots & R_{NN}^{KK} \end{pmatrix},$$

$$F = \begin{pmatrix} f_0^1 & f_1^1 & \dots & f_N^1 \\ f_0^2 & f_1^2 & \dots & f_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ f_0^{K-1} & f_1^{K-1} & \dots & f_N^{K-1} \end{pmatrix},$$

$$f_p^i = \begin{cases} I, p \leq N_i - 1, \text{ (or equivalently } i \geq M_{p+1}), \\ 0, \text{ otherwise,} \end{cases}$$

$$\bar{S} = \text{diag} \left( \frac{1}{h_{N_1}} S^1 \ \frac{1}{h_{N_2}} S^2 \ \dots \ \frac{1}{h_{N_{K-1}}} S^{K-1} \right),$$

$$\hat{S}' = \text{diag} \left( \frac{1}{h_0} S'_0 \ \frac{1}{h_1} S'_1 \ \dots \ \frac{1}{h_N} S'_N \right),$$

$$\tilde{h}_p = \max \{h_p, h_{p+1}\}, \quad p = 1, 2, \dots, N-1,$$

$$\tilde{h}_0 = h_1, \quad \tilde{h}_N = h_N,$$

$$S'_p = S_p^K + \sum_{i=M_{p+1}}^{K-1} S^i, \quad p = 0, 1, \dots, N.$$

*Proof.* The proof is very similar to that of Lemma 7.5 of Chap. 7 of [14].  $\square$

The next theorem established the conditions for (1.48). To derive the LKF derivative condition, it is noted that

$$\begin{aligned}\dot{Q}(\eta) &= \frac{1}{h_p}(Q_{p-1} - Q_p), \\ \dot{S}^K(\eta) &= \frac{1}{h_p}(S_{p-1}^K - S_p^K), \\ \dot{R}^{iK}(\eta) &= \frac{1}{h_p}(R_{p-1}^{iK} - R_p^{iK}) \\ &= \left( \frac{\partial}{\partial \xi} + \frac{\partial}{\partial \eta} \right) R^{KK}(\xi, \eta) \\ &= \begin{cases} \frac{1}{h_p}(R_{p-1,q-1}^{KK} - R_{p,q-1}^{KK}) + \frac{1}{h_q}(R_{p,q-1}^{KK} - R_{p,q}^{KK}), \alpha \leq \beta; \\ \frac{1}{h_q}(R_{p-1,q-1}^{KK} - R_{p-1,q}^{KK}) + \frac{1}{h_p}(R_{p-1,q}^{KK} - R_{p,q}^{KK}), \alpha > \beta; \end{cases}\end{aligned}$$

**Theorem 1.6.** *The derivative of the Lyapunov-Krasovskii functional  $\dot{V}$  in (1.44), with  $Q^K$ ,  $S^K$ ,  $R^{iK}$ , and  $R^{KK}$  piecewise linear as expressed in (1.49)–(1.52), satisfies (1.48) if*

$$\begin{pmatrix} \bar{\Delta} & Y^s & Y^a \\ R_{ds}^K + R_{ds}^{KK} + S_d^K - W & 0 & \\ \text{Symmetric} & 3(S_d - W) \end{pmatrix} > 0 \quad (1.56)$$

and

$$\begin{pmatrix} W & R_{ds}^{KK} \\ R_{ds}^{KK} & W \end{pmatrix} > 0, \quad (1.57)$$

where

$$\bar{\Delta} = \begin{pmatrix} \bar{\Delta}_{00} & \bar{\Delta}_{01} & \cdots & \bar{\Delta}_{0K} \\ \bar{\Delta}_{10} & \bar{\Delta}_{11} & \cdots & \bar{\Delta}_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\Delta}_{K0} & \bar{\Delta}_{K1} & \cdots & \bar{\Delta}_{KK} \end{pmatrix},$$

with  $\bar{\Delta}_{ij}, i = 0, 1, \dots, K, j = 0, 1, \dots, K$ , defined by (1.45) and (1.46);

$$Y^s = \begin{pmatrix} Y_{01}^s & Y_{02}^s & \cdots & Y_{0N}^s \\ Y_{11}^s & Y_{12}^s & \cdots & Y_{1N}^s \\ \vdots & \vdots & \ddots & \vdots \\ Y_{K1}^s & Y_{K2}^s & \cdots & Y_{KN}^s \end{pmatrix}, \quad (1.58)$$

$$\begin{aligned}
Y_{0p}^s &= \sum_{j=M_p}^K h_p (A^T Q^j + \sum_{l=1}^{K-1} C^T R^{lj} + C^T R_0^{jKT}) \\
&\quad + \frac{h_p}{2} \left[ A^T (Q_p^K + Q_{p-1}^K) + \sum_{l=1}^{K-1} C^T (R_p^{lK} + R_{p-1}^{lK}) \right. \\
&\quad \left. + C^T (R_{0p}^{KK} + R_{0,p-1}^{KK}) \right] + (Q_p^K - Q_{p-1}^K),
\end{aligned}$$

$$\begin{aligned}
Y_{ip}^s &= \sum_{j=M_p}^K h_p (B_i^T Q^j + D_i^T \sum_{l=1}^{K-1} R^{lj} + D_i^T R_0^{jKT} - R^{ij}) \\
&\quad + \frac{h_p}{2} \left[ B_i^T (Q_p^K + Q_{p-1}^K) + D_i^T \sum_{l=1}^{K-1} (R_p^{lK} + R_{p-1}^{lK}) \right. \\
&\quad \left. + D_i^T (R_{0p}^{KK} + R_{0,p-1}^{KK}) - (R_p^{iK} + R_{p-1}^{iK}) \right],
\end{aligned}$$

$$\begin{aligned}
Y_{Kp}^s &= \sum_{j=M_p}^K h_p (B_K^T Q^j + D_K^T \sum_{l=1}^{K-1} R^{lj} + D_K^T R_0^{jKT} - R_N^{iKT}) \\
&\quad + \frac{h_p}{2} \left[ B_K^T (Q_p^K + Q_{p-1}^K) + D_i^T \sum_{l=1}^{K-1} (R_p^{lK} + R_{p-1}^{lK}) \right. \\
&\quad \left. + D_i^T (R_{0p}^{KK} + R_{0,p-1}^{KK}) - (R_{Np}^{KK} + R_{N,p-1}^{KK}) \right];
\end{aligned}$$

$$Y^a = \begin{pmatrix} Y_{01}^a & Y_{02}^a & \cdots & Y_{0N}^a \\ Y_{11}^a & Y_{12}^a & \cdots & Y_{1N}^a \\ \vdots & \vdots & \ddots & \vdots \\ Y_{K1}^a & Y_{K2}^a & \cdots & Y_{KN}^a \end{pmatrix},$$

$$Y_{0p}^a = \frac{h_p}{2} \left[ A^T (Q_p^K - Q_{p-1}^K) + \sum_{l=1}^{K-1} C^T (R_p^{lK} - R_{p-1}^{lK}) + C^T (R_{0p}^{KK} - R_{0,p-1}^{KK}) \right],$$

$$\begin{aligned}
Y_{ip}^a &= \frac{h_p}{2} \left[ B_i^T (Q_p^K - Q_{p-1}^K) + D_i^T \sum_{l=1}^{K-1} (R_p^{lK} - R_{p-1}^{lK}) \right. \\
&\quad \left. + D_i^T (R_{0p}^{KK} - R_{0,p-1}^{KK}) - (R_p^{iK} - R_{p-1}^{iK}) \right],
\end{aligned}$$

$$\begin{aligned}
Y_{Kp}^a &= \frac{h_p}{2} \left[ B_K^T (Q_p^K - Q_{p-1}^K) + D_i^T \sum_{l=1}^{K-1} (R_p^{lK} - R_{p-1}^{lK}) \right. \\
&\quad \left. + D_i^T (R_{0p}^{KK} - R_{0,p-1}^{KK}) - (R_{Np}^{KK} - R_{N,p-1}^{KK}) \right];
\end{aligned}$$

$$S_d^K = \text{diag} \left( S_{d1}^K \ S_{d2}^K \ \dots \ S_{dN}^K \right),$$

$$S_{dp}^K = S_{p-1}^K - S_p^K, \quad 1 \leq p \leq N; \quad (1.59)$$

$$R_{ds}^K = \begin{pmatrix} R_{ds11}^K & R_{ds12}^K & \dots & R_{ds1N}^K \\ R_{ds21}^K & R_{ds22}^K & \dots & R_{ds2N}^K \\ \vdots & \vdots & \ddots & \vdots \\ R_{dsN1}^K & R_{dsN2}^K & \dots & R_{dsNN}^K \end{pmatrix},$$

$$R_{dspq}^K = \sum_{i=M_p}^{K-1} \left[ h_p(R_{q-1}^{iK} - R_q^{iK}) + h_q(R_{p-1}^{iKT} - R_p^{iKT}) \right];$$

$$R_{ds}^{KK} = \begin{pmatrix} R_{ds11}^{KK} & R_{ds12}^{KK} & \dots & R_{ds1N}^{KK} \\ R_{ds21}^{KK} & R_{ds22}^{KK} & \dots & R_{ds2N}^{KK} \\ \vdots & \vdots & \ddots & \vdots \\ R_{dsN1}^{KK} & R_{dsN2}^{KK} & \dots & R_{dsNN}^{KK} \end{pmatrix},$$

$$R_{dspq}^{KK} = \frac{1}{2} \left[ (h_p + h_q)(R_{p-1,q-1}^{KK} - R_{pq}^{KK}) + (h_p - h_q)(R_{p,q-1}^{KKT} - R_{p-1,q}^{KKT}) \right];$$

$$R_{da}^{KK} = \begin{pmatrix} R_{da11}^{KK} & R_{da12}^{KK} & \dots & R_{da1N}^{KK} \\ R_{da21}^{KK} & R_{da22}^{KK} & \dots & R_{da2N}^{KK} \\ \vdots & \vdots & \ddots & \vdots \\ R_{daN1}^{KK} & R_{daN2}^{KK} & \dots & R_{daNN}^{KK} \end{pmatrix},$$

$$R_{dapq}^{KK} = \frac{1}{2} (h_p - h_q) (R_{p-1,q-1}^{KK} - R_{p-1,q}^{KK} - R_{p,q-1}^{KK} + R_{pq}^{KK}). \quad (1.60)$$

*Proof.* Again, since the expression of  $\dot{V}$  is similar to that in [14] or [12], one may follow the same steps for the proof of Proposition 7.8 in [14]. See also [11, 24].  $\square$

From the above, the following can be concluded.

**Theorem 1.7.** *The system expressed by (1.39) and (1.40) is asymptotically stable if there exist  $m \times m$  matrix  $P = P^T$ ,  $m \times n$  matrices  $Q^i, Q_p^K$ , and  $n \times n$  matrices  $S^i = S^{iT}, S_p^K = S_p^{KT}, R^{ij} = R^{jiT}, R_p^{iK}, R_{pq}^{KK} = R_{qp}^{KKT}; i, j = 0, 1, \dots, K-1; p, q = 0, 1, \dots, N$ ; and  $Nn \times Nn$  matrix  $W = W^T$ , such that (1.55), (1.57), and*

$$\begin{pmatrix} \Delta & Y^s & Y^a & Z \\ & R_{ds}^K + R_{ds}^{KK} + S_d^K - W & 0 & 0 \\ & & 3(S_d - W) & 0 \\ \text{Symmetric} & & & \hat{S} \end{pmatrix} > 0 \quad (1.61)$$

are satisfied, where

$$\Delta = \begin{pmatrix} \Delta_{00} & \Delta_{01} & \cdots & \Delta_{0K} \\ \Delta_{10} & \Delta_{11} & \cdots & \Delta_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta_{K0} & \Delta_{K1} & \cdots & \Delta_{KK} \end{pmatrix}, \quad (1.62)$$

$$\begin{aligned} \Delta_{00} &= -[A^T P + P A + \sum_{l=1}^{K-1} (Q^l C + C^T Q^{lT}) + Q_0^K C + C^T Q_0^{KT}], \\ \Delta_{0j} &= -P B_j - \sum_{l=1}^{K-1} Q^l D_j + Q^j - Q_0^K D_j, \\ \Delta_{0K} &= -P B_K - \sum_{l=1}^{K-1} Q^l D_K + Q^K(-r) - Q_0^K D_K, \\ \Delta_{ij} &= 0, \quad 1 \leq i, j \leq K, \quad i \neq j, \\ \Delta_{ii} &= S^i, \quad 1 \leq i \leq K-1, \\ \Delta_{KK} &= S_N^K, \end{aligned} \quad (1.63)$$

and

$$Z = \begin{pmatrix} C^T \hat{S} \\ D_1^T \hat{S} \\ \vdots \\ D_K^T \hat{S} \end{pmatrix}, \quad (1.64)$$

$$\hat{S} = \sum_{i=1}^{K-1} S^i + S_0^K. \quad (1.65)$$

*Proof.* First, it is observed that (1.61) implies (1.53), (1.54), and (1.56). Furthermore, (1.61) also implies

$$\begin{pmatrix} S^1 & 0 & \cdots & 0 & 0 & D_1^T \hat{S} \\ 0 & S^2 & \cdots & 0 & 0 & D_2^T \hat{S} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & S^{K-1} & 0 & D_{K-1}^T \hat{S} \\ 0 & 0 & \cdots & 0 & S_0^K & D_K^T \hat{S} \\ \hat{S} D_1 & \hat{S} D_2 & \cdots & \hat{S} D_{K-1} & \hat{S} D_K & \hat{S} \end{pmatrix} > 0,$$

which is equivalent to

$$\text{diag}(S^1 \cdots S^{K-1} S_0^K) - D^T \left( \sum_{i=1}^{K-1} S^i + S_0^K \right) D > 0, \quad (1.66)$$

where  $D = (D_1 \ D_2 \ \cdots \ D_K)$ . Therefore, the subsystem (1.40) is uniformly input-to-state stable according to Theorem 1.3. Thus, all of the conditions in Theorem 1.4 has been established, and therefore, the system (1.39) and (1.40) is exponentially stable.  $\square$

Stability analysis based on the above theorem is known as the Discretized Lyapunov Functional Method. The idea can also be extended to uncertain systems. Indeed, let

$$\omega = (A, B_1, B_2, \dots, B_K, C, D_1, D_2, \dots, D_K).$$

If the system matrices  $\omega$  is not known exactly, but is known to be within a bounded closed set  $\Omega$ , then the system is obviously asymptotically stable provided that Theorem 1.7 is satisfied for all  $\omega \in \Omega$ . Indeed, this is also true for  $\omega = \omega(t)$  provided that  $D_i$ ,  $i = 1, 2, \dots, K$  are independent of time. If  $\Omega$  is polytopic, then let its vertices be  $\omega_k$ ,  $k = 1, 2, \dots, n_v$ , then  $\Omega$  is the convex hull of  $\{\omega_k, k = 1, 2, \dots, n_v\}$ , and the conditions in Theorem 1.7 only need to be satisfied in the  $n_v$  vertices  $\omega_k$ ,  $k = 1, 2, \dots, n_v$ . The idea is very similar to the corresponding results for retarded time-delay systems discussed in [14]. Notice, however, that the case of time-varying  $D_i$  is excluded as the corresponding stability results for the difference equations have not yet been established.

## 1.5 Discussion and Examples

Many practical systems have a high dimension, but only a small number of the elements have time delays. In this case, the coupled DDE formulation offers a significant advantage over the traditional model even for time-delay systems of the retarded type. Indeed, as suggested in the introduction, if one rewrites the system by “pulling out the delays,” one arrives as a system with forward system described by

$$\dot{x}(t) = Ax(t) + Bu(t), \quad (1.67)$$

$$y(t) = Cx(t) + Du(t), \quad (1.68)$$

and the feedback system consisting of pure delays

$$u(t) = \sum_{i=1}^K F_i y(t - r_i), \quad i = 1, 2, \dots, K., \quad (1.69)$$

where  $x(t) \in \mathbb{R}^m$ ,  $y(t) \in \mathbb{R}^n$ , and  $u(t) \in \mathbb{R}^n$ . Obviously, a substitution of (1.67) and (1.68) by (1.69) results in a standard form of coupled DDEs (1.39) and (1.40). Typically, for a large system with a small number of delay elements,  $m \gg n$ . The special case of  $D_i = 0$  ( $i = 1, 2, \dots, k$ ) can also be written as

$$\dot{x}(t) = Ax(t) + \sum_{i=1}^K BF_i Cx(t - r_i), \quad (1.70)$$

which is a time-delay system of the retarded type. Even in this case, the model (1.39) and (1.40) offers a significant advantage over (1.70) in stability analysis using the Discretized Lyapunov Functional Method. Indeed, the dimensions of LMIs (1.55), (1.61), and (1.57) applicable to (1.39) and (1.40) are  $m + (N + K)n$ ,  $m + (2N + K + 1)n$ , and  $2Nn$ , respectively, which are much smaller than  $(K + N + 1)m$ ,  $(K + 2N + 1)m$ , and  $2Nm$ , the dimensions of LMIs (20), (27), and (28) in [12] applicable to (1.70).

In the remaining part of this chapter, three numerical examples are presented to illustrate the effectiveness of the method. All numerical calculations were carried out by using the LMI Toolbox [10] of MATLAB 7.0 on a laptop PC with a 1.86 GHz processor and with 512 MB RAM. The time is measured in seconds. It is also defined that

$$N_{di} = N_i - N_{i-1}.$$

*Example 1.1.* Consider the system

$$\ddot{x}(t) - 0.1\dot{x}(t) + x(t) + x(t - r/2) - x(t - r) = 0.$$

The above example is studied in [12] and it is shown that the system is unstable for  $r = 0$ . For small  $r$ , as a finite difference, the last two terms can approximate  $r\dot{x}(t)/2$ , and as  $r$  increases, it may improve stability. In [12], the system is written in the state-space form

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0.1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t - r/2) \\ x_2(t - r/2) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t - r) \\ x_2(t - r) \end{pmatrix}. \quad (1.71)$$

It is known that for  $r = 0.2025$ , the system has a pair of imaginary poles at  $\pm 1.0077i$ . For  $r = 1.3723$ , the system has a pair of imaginary poles  $\pm 1.3386i$ . The system is stable for  $r \in (0.2025, 1.3723)$ . In [12], the discretized Lyapunov-Krasovskii functional method was used to estimate the delay interval such that the system remains stable, and it was found that  $[r_{\min}, r_{\max}] = [0.204, 1.350]$  for  $N_{d1} = N_{d2} = 1$ , and  $[r_{\min}, r_{\max}] = [0.203, 1.372]$  for  $N_{d1} = N_{d2} = 2$ . Here, the system is written as an equivalent couple different-difference equation with two delays

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0.1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} - \begin{pmatrix} 0 \\ 1 \end{pmatrix} y(t - r/2) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} y(t - r), \quad (1.72)$$

$$y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}. \quad (1.73)$$

It can be verified that the dimensions of LMIs (1.55), (1.61), and (1.57) applicable to (1.72) and (1.73) are  $N + 4$ ,  $2N + 6$ , and  $2N$ , respectively, which are much smaller than  $2N + 6$ ,  $4N + 6$ , and  $4N$ , the dimensions of LMIs (20), (27), and (28) in [12] applicable to (1.71). A bisection process was used. An initial interval containing  $r_{\max}$  or  $r_{\min}$  of length 2 is subdivided 15 times until the interval size is less than  $6.135 \times 10^{-5}$ . The estimated results for  $N_{d1} = N_{d2} = 1, 2, 3$  are listed in the

following table. The numerical result is virtually identical, which was achieved with substantially reduced computational cost.

$(N_{d1}, N_{d2})$	(1, 1)	(2, 2)	(3, 3)
$r_{\min}$	0.2032	0.2026	0.2025
Time	29.6875	291.1875	1,236.9
$r_{\max}$	1.3246	1.3719	1.3723
Time	29.2813	226.1094	630.0469

*Example 1.2.* Consider the following system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B_1(t)y(t - 2r) + Bu(t), \\ y(t) &= Cx(t),\end{aligned}$$

where

$$\begin{aligned}A &= \begin{pmatrix} -\frac{61}{2} & 1 & 0 & 0 \\ -200 & 0 & 1 & 0 \\ -305 & 0 & 0 & 1 \\ -100 & 0 & 0 & 0 \end{pmatrix}, \\ B_1 &= \begin{pmatrix} 0 \\ 0 \\ -200 \\ -250 \end{pmatrix}, \\ B &= \begin{pmatrix} 0 & 0 \\ 20 & 20 \\ 305 & 105 \\ 350 & 100 \end{pmatrix}, \\ C &= (1 \ 0 \ 0 \ 0).\end{aligned}$$

This system represents the model for a tele-operator robotics system studied in [33] where the transfer function was used.

It is supposed the following feedback control is applied

$$u(t) = \begin{pmatrix} \rho & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma \end{pmatrix} x(t) - \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} y(t - \sqrt{2}r),$$

where  $\rho, \sigma$  are time-invarying parameters satisfying

$$|\rho| \leq \rho_0, \quad |\sigma| \leq \sigma_0.$$

Again, the formulation in this chapter is applied with  $N_{d1} = N_{d2} = 2$ , and the results are listed in the following table. From the table, it is observed that the uncertainties significantly affect the maximum delay allowed for the system to remain stable.

$(\rho_0, \sigma_0)$	(0,0)	(0,0.015)	(0,0.018)
$r_N$	0.2209	0.0309	0.0060
Time	597.8594	1,159.9	1,423.8
$(\rho_0, \sigma_0)$	(0.1, 0)	(0.50, 0)	(1.00, 0)
$r_N$	0.1957	0.1149	0.0283
Time	1,160.4	1,238	1,229.8
$(\rho_0, \sigma_0)$	(0.1, 0.01)	(0.4, 0.01)	(0.4, 0.011)
$r_N$	0.0635	0.0134	0.0036
Time	3,087	2,927.2	3,053.2

The above two examples are of retarded type. The following example is of neutral type.

*Example 1.3.* Consider the neutral time-delay system

$$\begin{aligned} & \frac{d}{dt} [x(t) - 0.8x(t-s) - 0.4x(t-r) + 0.32x(t-s-r)] \\ &= -0.848x(t) + 0.72x(t-s) + 0.128x(t-s-r). \end{aligned}$$

This example is similar to the example discussed on page 289 of [22]. It is important to observe that the three delays depend on two independent parameters, and the dependence is guaranteed by the structure, and is not subject to any small deviations even though the parameters  $r$  and  $s$  may contain small errors. It is therefore important to transform the system to a form such that only  $r$  and  $s$  appear as delays in view of the sensitivity of difference equations to small delays. Let

$$\begin{aligned} z(t) &= x(t) - 0.8x(t-s) - 0.4x(t-r) + 0.32x(t-s-r), \\ y(t) &= \begin{pmatrix} x(t) \\ x(t-r) \end{pmatrix}. \end{aligned}$$

Then the system can be written in a standard form with two independent delays

$$\begin{aligned} \dot{z}(t) &= -0.848z(t) + \begin{pmatrix} 0.0416 & 0.3994 \end{pmatrix} y(t-s) + \begin{pmatrix} -0.3392 & 0 \end{pmatrix} y(t-r), \\ y(t) &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} z(t) + \begin{pmatrix} 0.8 & -0.32 \\ 0 & 0 \end{pmatrix} y(t-s) + \begin{pmatrix} 0.4 & 0 \\ 1 & 0 \end{pmatrix} y(t-r). \end{aligned}$$

Let

$$\mathcal{D}(\phi) = \phi(0) - D_1 \phi(-s) - D_2 \phi(-r)$$

be the difference operator, where

$$D_1 = \begin{pmatrix} 0.8 & -0.32 \\ 0 & 0 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 0.4 & 0 \\ 1 & 0 \end{pmatrix}.$$

After using Theorem 1.3, one may conclude that the difference equation  $\mathcal{D}(\phi)$  is stable. Indeed, the matrices

$$S^1 = \begin{pmatrix} 2.3051 & -0.9730 \\ -0.9730 & 0.7110 \end{pmatrix},$$

$$S^2 = \begin{pmatrix} 0.9578 & -0.2109 \\ -0.2109 & 0.4055 \end{pmatrix},$$

satisfy

$$\begin{pmatrix} S^1 & 0 \\ 0 & S^2 \end{pmatrix} - \begin{pmatrix} D_1^T \\ D_2^T \end{pmatrix} (S^1 + S^2) (D_1 \ D_2) > 0.$$

Next, the allowable maximum  $r$  is estimated for the system to retain stability if it is let that  $s = cr$ . The discretized Lyapunov-Krasovskii functional method with  $N_{d1} = N_{d2} = 2$  is used, and the results are listed for different  $c$ .

$c$	$\sqrt{2}$	$\sqrt{7}$	3	$\sqrt{13}$
$r_N$	2.2874	2.8759	0.7318	0.5977
Time	2,029.8	1,768.9	1,606.8	1,740

## 1.6 Concluding Remarks

A Lyapunov-Krasovskii functional approach of systems described by coupled DDEs is presented in this chapter. For linear systems, a discretization is applied to render the condition in the form of linear matrix inequality. Even in the case of retarded delay, the method presented offers significant advantages over the existing method as the dimension of the LMI system is significantly lower.

It is interesting to discuss a number of possible alternatives. In the Lyapunov-Krasovskii functional approach, as mentioned earlier, an alternative formulation with one delay in each  $y_i$  was proposed in [13] with more complete theoretical results. The corresponding discretized Lyapunov functional method is presented in [34]. It is also conceivable that other methods of reducing the quadratic Lyapunov-Krasovskii functional to a finite dimensional LMI, such as the sum-of-square formulation pursued in [40], or alternative discretization pursued in [37, 38], can also be formulated.

In the frequency-domain approach, typical results are in the form of “stability charts” describing the parameter regions in which the system is stable [51]. Some examples of these charts can be found in Chaps. 4 and 5. As these results are based on the characteristic quasipolynomial of the system, the results obtained for time-delay systems of neutral type apply equally well to coupled DDEs. At present, a number of methods are well established for systems with single delay or commensurate delays [39, 53]. For two incommensurate delays *in the characteristic*

*quasipolynomial*, a very systematic method is available in [18]. Unfortunately, this does not solve the problem of systems with two incommensurate delays  $\tau_1$  and  $\tau_2$  for general linear systems, as the resulting characteristic quasipolynomial is in the form of

$$\sum_{k=1}^K p(s)e^{-r_k s}, \quad (1.74)$$

where  $p(s)$  are polynomials, and  $r_k, k = 1, 2, \dots, K$  are integer combinations of  $\tau_1$  and  $\tau_2$ . It is interesting to mention that [4] discussed the stability of systems with the characteristic quasipolynomial

$$p(s) + q(s)e^{-rs}, \quad (1.75)$$

where  $p(s)$  and  $q(s)$  are quasipolynomials. This gives the possibility of solving the stability problem of (1.74) by considering one  $r_k$  at a time. However, the computational implementation is not easy as it requires a good knowledge of, for example, the zero distribution of the coefficient quasipolynomials along the imaginary axis. A more recent interesting paper on the stability problem of (1.74) is [8], even though the procedure is far from systematic and not easy to implement for a general case.

**Acknowledgments** This work was completed while Dr. Hongfei Li was visiting Southern Illinois University Edwardsville. Dr. Li's work was supported by the Natural Science Foundation of Shaanxi Province, China (Grant number 2006A13) and the Foundation for Scientific Research, Department of Education of Shaanxi Province, China (Grant number 06JK149), and the State Scholarship Fund of China.

## References

1. Boyd, S., El Ghaoui, L., Feron E., & Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*. Philadelphia: SIAM.
2. Brayton, R. (1976). Nonlinear oscillations in a distributed network. *Quarterly of Applied Mathematics*, 24, 289–301.
3. Carvalho, L. A. V. (1996). On quadratic Liapunov functionals for linear difference equations. *Linear Algebra and its Applications*, 240, 41–64.
4. Cooke, K. L., & van den Driessche, P., 1986. On zeros of some transcendental equations. *Funkcialaj Ekvacioj*, 29, 77–90.
5. Cruz, M. A., & Hale, J. K. (1970). Stability of functional differential equations of neutral type. *Journal of Differential Equations*, 7, 334–355.
6. Doyle, J. C. (1982). Analysis of feedback systems with structured uncertainties. *IEEE Proceedings Part D*, 129(6), 242–250.
7. Doyle, J. C., Wall, J., & Stein, G. (1982). Performance and robustness analysis for structured uncertainty. *20th IEEE Conference on Decision and Control*, 629–636.
8. Fazelinia, H., Sipahi, R., & Olgac, H. (2007). Stability robustness analysis of multiple time-delayed systems using “building block” concept. *IEEE Transactions on Automatic Control*, 52(5), 799–810.
9. Fridman, E. (2002). Stability of linear descriptor systems with delay: a Lyapunov-based approach. *Journal of Mathematical Analysis and Applications*, 273, 14–44.

10. Gahinet, P., Nemirovski, A., Laub, A., & Chilali, M. (1995). *LMI control toolbox for use with MATLAB*. Natick, MA: Mathworks.
11. Gu, K. (2001). A further refinement of discretized Lyapunov functional method for the stability of time-delay systems. *International Journal of Control*, 74(10), 967–976.
12. Gu, K. (2003). Refine discretized Lyapunov functional method for systems with multiple delays. *International Journal of Robust Nonlinear Control*, 13, 1017–1033.
13. Gu, K. (2008). Large systems with multiple low-dimensional delay channels. Semi-plenary lecture, *Proceedings of 2008 Chinese Control and Decision Conference*, Yantai, China, July 2–4.
14. Gu, K., Kharitonov, V. L., & Chen, J. (2003). *Stability of Time-Delay Systems*. Boston: Birkhäuser.
15. Gu, K., & Liu, Y. (2007). Lyapunov-Krasovskii functional for coupled differential-functional equations. *46th Conference on Decision and Control*, New Orleans, LA, December 12–14.
16. Gu, K., & Liu, Y. (2008). Lyapunov-Krasovskii functional for uniform stability of coupled differential-functional equations. Accepted for publication in *Automatica*.
17. Gu, K., & Niculescu, S.-I. (2006). Stability analysis of time-delay systems: A Lyapunov approach. In: Loria, A., Lamnabhi-Lagarigue, F., & Panteley, E. (Eds.), *Advanced Topics in Control Systems Theory*, Lecture Notes from FAP 2005 (pp. 139–170). London: Springer.
18. Gu, K., Niculescu, S.-I., & Chen, J. (2005). On stability of crossing curves for general systems with two delays. *Journal of Mathematical Analysis and Applications*, 311, 231–253.
19. Hale, J. K. (1975). Parametric stability in difference equations. *Bollettino Univerdella Matematica Italiana*, 4, Suppl. 11.
20. Hale, J. K., & Cruz, M. A. (1970). Existence, uniqueness and continuous dependence for hereditary systems. *Annali di Matematica Pura ed Applicata*, 85(1), 63–81.
21. Hale, J., & Huang, W. (1993). Variation of constants for hybrid systems of functional differential equations. *Proceedings of Royal Society of Edinburgh*, 125A, 1–12.
22. Hale, J. K., & Verduyn Lunel, S. M. (1993). *Introduction to functional differential equations*. New York: Springer.
23. Hale, J. K., & Verduyn Lunel, S. M. (2002). Strong stabilization of neutral functional differential equations. *IMA Journal of Mathematical Control and Information*, 19, 5–23.
24. Han Q., & Yu, X. (2002). A discretized Lyapunov functional approach to stability of linear delay-differential systems of neutral type. *Proceedings of the 15th IFAC World Congress on Automatic Control*, Barcelona, Spain, July 21–26, 2002.
25. Jiang, Z.-P. & Wang, Y. (2001). Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37, 857–869.
26. Kabaov, I. P. (1946). On steam pressure control [Russian]. *Inzhenerno Sbornik*, 2, 27–46.
27. Karafyllis, I., Pepe, P., & Jiang, Z.-P. (2007). Stability results for systems described by coupled retarded functional differential equations and functional difference equations. *Seventh Workshop on Time-Delay Systems*, Nantes, France, September 17–19.
28. Karafyllis, I., Pepe, P., & Jiang, Z.-P. (2007). Stability Results for Systems Described by Retarded Functional Differential Equations. *2007 European Control Conference*, Kos, Greece, July 2–5.
29. Karaev, R. I. (1978). *Transient processes in long distance transmission lines* [Russian]. Moscow: Energia Publishing House.
30. Kharitonov, V. L., & Zhabko, A. P. (2003). Lyapunov-Krasovskii approach to the robust stability analysis of time-delay systems. *Automatica*, 39, 15–20.
31. Kolmanovskii V., & Myshkis, A. (1999). *Introduction to the theory and applications of functional differential equations*. Dordrecht, the Netherlands: Kluwer.
32. Krasovskii, N. N. (1959). *Stability of motion* [Russian], Moscow [English translation, 1963]. Stanford, CA: Stanford University Press.
33. Lee, S., & Lee, H. S. (1993). Modeling, design, and evaluation of advanced teleoperator control systems with short time delay. *IEEE Transactions on Robotics and Automation*, 9(5), 607–623.

34. Li, H., & Gu, K. (2008). Discretized Lyapunov-Krasovskii functional for systems with multiple delay channels. In the Invited Session "Time-Delay Systems: From Theory to Applications," 2008 ASME Dynamic Systems and Control Conference, Ann Arbor, MI, October 20–22.
35. Martinez-Amores, P. (1979). Periodic solutions of coupled systems of differential and difference equations. *Annali di Matematica Pura ed Applicata*, 121(1), 171–186.
36. Niculescu, S.-I. (2001). *Delay effects on stability—A robust control approach*, Lecture Notes in Control and Information Science, 269. London: Springer.
37. Ochoa, G., & Kharitonov, V. L. (2005). Lyapunov matrices for neutral type of time delay systems. *Second International Conference on Electrical & Electronics Engineering and Eleventh Conference on Electrical Engineering*, Mexico City, September 7–9.
38. Ochoa, G., & Mondie, S. (2007). Approximations of Lyapunov-Krasovskii functionals of complete type with given cross terms in the derivative for the stability of time-delay systems. *46th Conference on Decision and Control*, New Orleans, LA, December 12–14.
39. Olgac, N., & Sipahi, R. (2002). An exact method for the stability analysis of time-delayed LTI systems. *IEEE Transactions on Automatic Control*, 47(5): 793–797.
40. Peet, M., Papachristodoulou A., & Lall, S. (2006). On positive forms and the stability of linear time-delay systems. *45th Conference on Decision and Control*, San Diego, CA, December 13–15.
41. Pepe, P., & Verriest, E. I. (2003). On the stability of coupled delay differential and continuous time difference equations. *IEEE Transactions on Automatic Control*, 48(8), 1422–1427.
42. Pepe, P. (2005). On the asymptotic stability of coupled delay differential and continuous time difference equations. *Automatica*, 41(1), 107–112.
43. Pepe, P., Jiang, Z.-P., & Fridman, E. (2007). A new Lyapunov-Krasovskii methodology for coupled delay differential and difference equations. *International Journal of Control*, 81(1), 107–115.
44. Rasvan, V. (1973). Absolute stability of a class of control processes described by functional differential equations of neutral type. In: Janssens, P., Mawhin, J., & Rouche N. (Eds), *Equations Differentielles et Fonctionnelles Nonlineaires*. Paris: Hermann.
45. Răşvan, V. (1998). Dynamical systems with lossless propagation and neutral functional differential equations. *Proceedings of MTNS 98*, Padoue, Italy, 1998, 527–531.
46. Răşvan, V. (2006). Functional differential equations of lossless propagation and almost linear behavior, Plenary Lecture. *Sixth IFAC Workshop on Time-Delay Systems*, L'Aquila, Italy.
47. Răşvan, V., & Niculescu, S.-I. (2002). Oscillations in lossless propagation models: a Liapunov-Krasovskii approach. *IMA Journal of Mathematical Control and Information*, 19, 157–172.
48. Silkowskii, R. A. (1976). *Star-shaped regions of stability in hereditary systems*. Ph.D. thesis, Brown University, Providence, RI, June.
49. Sontag, E. D. (1989). Smooth stabilization implies coprime factorization. *IEEE Transactions on Automatic Control*, 34, 435–443.
50. Sontag, E. D. (1990). Further facts about input to state stabilization. *IEEE Transactions on Automatic Control*, 35, 473–476.
51. Stépán, G. (1989). *Retarded dynamical systems: stability and characteristic function*. New York: Wiley.
52. Teel, A. R. (1998). Connections between Razumikhin-type theorems and the ISS nonlinear small gain theorem. *IEEE Transactions on Automatic Control*, 43(7), 960–964.
53. Walton, K., & Marshall, J. E. (1987). Direct Method for TDS Stability Analysis. *IEE Proceedings*, Part D, 134(2), 101–107.

## Chapter 2

# Networked Control and Observation for Master–Slave Systems

Wenjuan Jiang, Alexandre Kruszewski, Jean-Pierre Richard,  
and Armand Toguyeni

**Abstract** This chapter concerns the design of a remote control loop that combines a Slave system (with computing and energy limitations) and a Master computer, communicating via an Internet connection. In such a situation, the communication cost is reduced but the quality of service (QoS) of the Internet connection is not guaranteed. In particular, when the Slave dynamics are expected to be fast enough, the network induces perturbations (delays, *jitters*, packet dropouts and sampling) that may damage the performance. Here, the proposed solution relies on a delay-dependent, state-feedback control computed by the Master on the basis of an observer. This last estimates the present Slave's state from its past sampled outputs, despite the various delays. Then, the computing task is concentrated in the Master. The theoretical results are based on the Lyapunov-Krasovskii functional and the approach of LMI, which guarantee the stabilization performance with respect to the expected maximum delay of the connection. Two strategies are applied: one is a constant controller/observer gain strategy, which takes into account a fixed upperbound for the communication delay. The second strategy aims at improving the performance by adapting the gains to the available network QoS (here, with two possible upperbounds).

**Keywords:** Remote control · Switching signal · Exponential stability · Linear time-delay system · LMIs · Internet · UDP · Robot

## 2.1 Introduction

A networked control system (NCS) is a type of closed-loop control system with real-time communication networks imported into the control channel and feedback channel. The control and feedback signals in a NCS exchanged among the system's components are in the form of information packets through a network. The network-induced delay is brought into the control systems, which may create unstable behaviors.

Network induced delays vary depending on the network hardware, on the different protocols, on the traffic load, etc. In some cases, such as in a token ring local area network, the time-delay is bounded; for other networks such as Ethernet and Internet, the time-delay is unbounded and varying.

As Internet and Ethernet are well developed, remote control systems have been widely used in industrial, communicational, and medical systems, to cite a few. However, alongside the advantage of low costs, the Internet inevitably brings problems to a closed-loop controlled system, such as delay variation, data-packets loss [1] and disorder, which can cause poor performance, instability, or danger (see for instance Chap. 5 of [2, 3] and the references therein).

How to diminish the effect of time delay in the remote system is critical in the system design. The main solution can split into two (combinable) strategies [2, 4]: (1) Increase the network performances quality of service or (2) design an adapted control that can compensate the network influence. In this chapter, we consider this last approach for the network controlled system via Internet or Ethernet. The experiments we propose, in the last part, use the Internet, but the control strategy holds for both network standards.

A variety of stability and control techniques have been developed for general time delay systems [5–8]. Applications of these techniques to NCS were also derived [1, 9–14]. But some of these results are based on simplifying assumptions (for instance, the delays are constant) or lead to technical solutions that decrease the performance (for instance, a “buffer strategy” allows the communication delays to become constant by waiting enough after the data are received). In fact, to consider the time delay as constant [10, 15–17] is actually unrealistic because of the dynamic character of the network. A delay maximizing strategy [8, 11] (“virtual delay,” “buffer,” or “waiting” strategy) can be carried out so as to make the delay constant and known. This requires the knowledge of the maximum delay values  $h_m$ . However, it is obvious that maximizing the delay up to its largest value decreases the speed performance of the remote system. Several results are limited to a time-delay, whose value is less than the sensor and controller sampling periods [18]. In the Internet case, this constraint leads to an increase of the sampling periods up to the maximal network delay, which may be constraining for high dynamic applications.

Note that, in the Internet case, the network delays cannot be modeled or predicted. Moreover, the (variable) transmission delays are asymmetric, which means that the delay  $h_1(t)$  from Master to Slave (shortly, M-to-S), and the return one (S-to-M)  $h_2(t)$  normally satisfy  $h_1(t) \neq h_2(t)$ . Because of this lack of knowledge, predictor-based control laws [12] cannot be applied.

Our aim is to ensure suitable stabilization and speed performances, i.e. exponential stabilization, despite the dynamic variations of the network.

Our solution relies on the theoretical results of [19] (exponential stabilization of systems with unknown, varying delays), as well as [13] (networked control), the main lines of which will be given in the next section. It allows one to apply a waiting strategy only to the M-to-S communication, whereas the S-to-M communication takes the sensor measurements into account as soon as received. In order to enhance the performance of the system, a gain scheduling strategy is adapted

according to the variable time-delay of the network involved. In our application, we use Internet for a long-distance remote control. The Master-Slave system is based on the UDP protocol and involves lists as buffers. The choice of UDP is preferred to TCP because in our NCS situation re-emitting packets is not needed and setting up the TCP connection between two PCs is time-consuming.

## 2.2 Exponential Stability of a Remote System Controlled Through Internet

We consider the remote system based on the Master-Slave structure. For energy saving reasons, the work of the Slave PC is simplified, while the control and observation complexity is concentrated on the Master. For the main features of the system, refer to Fig. 2.1.

### 2.2.1 The Three Delay Sources

In such a situation, the variable delays come from (1) the communication through the Internet, (2) the data-sampling; and (3) the possible packet losses. In the sequel,  $h_1(t)$  and  $h_2(t)$  denote the communication delays and  $\tau_1(t)$  and  $\tau_2(t)$  include the sampling delays and possible packet losses. The total delay  $\delta_i(t)$  between Master and Slave results from the addition of  $h_i(t)$  and  $\tau_i(t)$  for  $i \in \{1, 2\}$ .

1. Both computers' dates are automatically synchronized in the system. The strategy of NTP (Network Time Protocol) [20] is used in the program to calculate the time clock difference between the two sides. By this way, whenever the Master or the Slave receives the data, including the time stamp, it knows the instant  $t_k$  of data sending out and the resulting transmission delay  $h_i(t_k)$ ,  $i = 1, 2$ .
2. The real remote system, including Master, Slave, and network, must involve some data sampling. However, following [21, 22], this phenomenon is equivalent to a time-varying, discontinuous delay  $\tau_i(t)$  (defined in Section (2.1)), which allows one to keep a continuous-time model. If the packets exchange between

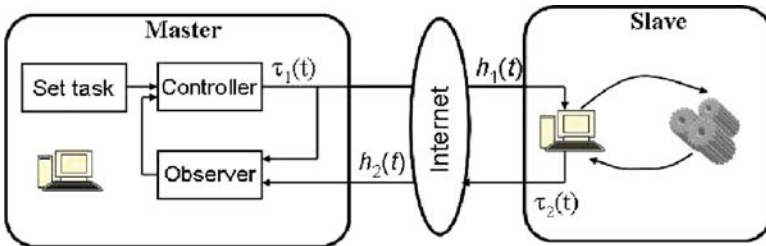


Fig. 2.1 Structure of general M/S based remote system

the Master and the Slave is of high speed, then  $\tau_i(t)$  constitutes a disturbance that should be considered in the stabilization design [1].  $\tau_i(t)$  is variable but it is assumed there is a known  $T$  (maximum sampling period) so that  $\tau_i(t) \leq T$ .

3. If some packet  $p_{t_k}$ , containing the sample at  $t_k$ , is lost, or arrives later than the packet  $p_{t_{k+1}}$ , then the Master only considers the most recent data (i.e., those from  $p_{t_{k+1}}$ ). If it is assumed that the maximum number of successive packets that can be lost is  $N$ , then the maximum resulting delay is  $NT$ . The same conditions hold for the control packets.

From assumptions (2) and (3) above, the delay  $\delta_i(t)$  has a known maximum  $\delta_i^m(t) = (N+1)T + h_m$  and the delay variation satisfies  $\dot{\delta}_i(t) \leq 1$ . To keep simple expressions, the notation  $T$  will be kept preferable to  $T' = (N+1)T$ .

Summarizing, given a signal  $g(t)$  and the global delay  $\delta(t)$ , which represents the combination of the sampling and packet delay  $h(t_k)$  that the transmission line imposes on the packet containing the  $k^{\text{th}}$  sample at time  $t_k$ , then  $g(t)$  can be written as:

$$\begin{aligned} g(t_k - h(t_k)) &= g(t - h(t_k) - (t - t_k)), \\ &= g(t - \delta(t)), \\ t_k \leq t < t_{k+1}, \quad \delta(t) &\triangleq h(t_k) + \tau_k(t), \\ \tau_k(t) &= t - t_k. \end{aligned} \tag{2.1}$$

## 2.2.2 Transmission and Receipt of the Control Data

The  $k^{\text{th}}$  data sent by the Master to Slave includes the control  $u(t_{1,k})$  together with the time stamp when the packet is sent. At time  $t_{1,k}^r$ , when the Slave receives the data it can calculate the time delay because of the time stamp. If the delay equals  $h_{1m}$ , the Slave should apply immediately the control command.

The control  $u$ , sent out by the Master at time  $t_{1,k}$ , is received by the Slave at time  $t_{1,k}^r > t_{1,k}$ . It will be injected into the Slave input only at the predefined “target time”  $t_{1,k}^{\text{target}} = t_{1,k} + h_{1m}$ . The corresponding waiting time  $t_{1,k} + h_{1m} - t_{1,k}^r$  is depicted in Fig. 2.2. This is realistic because the transmission delay is bounded by a known value  $h_{1m}$ . By this way, the Master knows the time  $t_{1,k} + h_{1m}$  when this control  $u(t_{1,k})$  will be injected into the Slave input.

## 2.2.3 Problem Formulation and Preliminaries

Consider the Slave as a linear system. It is described in the following form, in which  $(A, B, C)$  is controllable and observable.

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t - \delta_1(t)), \\ y(t) = Cx(t), \end{cases} \tag{2.2}$$

where  $\delta_1(t) = \delta_1 + \eta_1(t)$ ,  $\|\eta_1(t)\| \leq \mu_1$ .

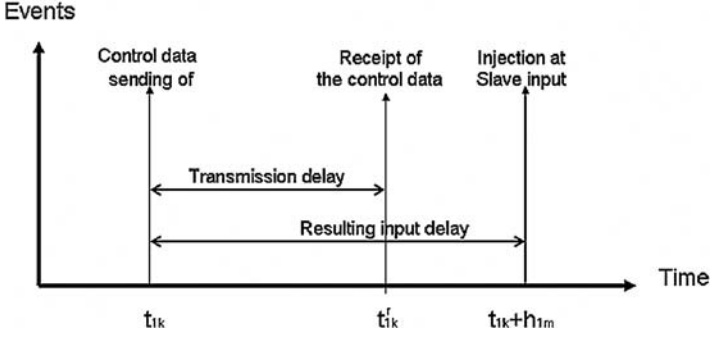


Fig. 2.2 Control data processing

In order to guarantee the closed-loop stability, whatever the delay variation, an exponential stability, with the rate  $\alpha$ , must be achieved. In other words, there must be a real  $\kappa \geq 1$  so that the solution  $x(t; t_0, \phi)$ , starting at any time  $t_0$ , from any initial function  $\phi$  satisfies:  $\|x(t; t_0, \phi)\| \leq \kappa \|\phi\| e^{-\alpha(t-t_0)}$ . In this chapter, it is achieved using a state observer and a state feedback.

To ensure this exponential global stabilization, one can use the results of [13], which considers a Lyapunov-Krasovskii functional with descriptor representation:

$$V(t) = \bar{x}_\alpha^T(t) E P \bar{x}_\alpha(t) + \int_{-\delta_1}^0 \int_{t+\theta}^t \dot{\bar{x}}_\alpha^T(s) R \dot{\bar{x}}_\alpha(s) ds d\theta + \int_{t-\delta}^t \bar{x}_\alpha^T(s) S x_\alpha(s) ds + \int_{-\mu_1}^{\mu_1} \int_{t+\theta-\delta_1}^t \dot{\bar{x}}_\alpha^T(s) R_a \dot{\bar{x}}_\alpha(s) ds d\theta, \quad (2.3)$$

where  $\bar{x}_\alpha(t) = \text{col}\{x_\alpha(t), \dot{x}_\alpha(t)\}$ ,  $x_\alpha(t) = x(t)e^{\alpha t}$ , and  $E = \text{diag}\{I, 0_{(2 \times 2)}\}$ .

Because of the separation principle, one can divide the analysis of the global stabilization into two smaller problems: the observer design and the controller design. The results are recalled below using an observer/controller.

### 2.2.4 Observer Design

For a given  $k$  and for any  $t \in [t_{1,k} + h_{1m}, t_{1,k+1} + h_{1m}]$ , there exists a  $k'$  such that the proposed observer is of the form:

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t_{1,k}) - L(y(t_{2,k'}) - \hat{y}(t_{2,k'})), \\ \hat{y}(t) = C\hat{x}(t). \end{cases} \quad (2.4)$$

The index  $k'$  corresponds to the most recent output information that the Master has received. Note that the Master knows the time  $t_{1,k}$  and the control  $u(t_{1,k})$  (see Sect. 2.3.3), which makes this observer realizable.

Using the delay (2.1) rewrite (2.4) as:

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t - \delta_1(t)) - L(y(t - \delta_2(t)) - \hat{y}(t - \delta_2(t))), \\ \hat{y}(t) = C\hat{x}(t). \end{cases} \quad (2.5)$$

with  $\delta_1(t) \triangleq t - t_{1,k}$  and  $\delta_2(t) \triangleq t - t_{2,k'}$ . In other words, the observer is realizable because the times  $t_{1,k}$  and  $t_{2,k'}$  defining the observer delays are known, thanks to the time stamps. The system features lead to  $\delta_1(t) \leq h_{1m} + T$  and  $\delta_2(t) \leq h_{2m} + T$ .

We define the error vector between the estimated state  $\hat{x}(t)$  and the present system state  $x(t)$  as  $e(t) = x(t) - \hat{x}(t)$ . From (2.2) and (2.5), this error is given by:

$$\dot{e}(t) = Ae(t) - LCe(t - \delta_2(t)). \quad (2.6)$$

**Theorem 2.1.** [13] Suppose that, for some positive scalars  $\alpha$  and  $\varepsilon$ , there exists  $n \times n$  matrices  $0 < P_1, P, S, Y_1, Y_2, Z_1, Z_2, Z_3, R, R_a$  and a matrix  $W$  with appropriate dimensions such that the following LMI conditions are satisfied for  $j = 1, 2$ :

$$\begin{bmatrix} \Psi_2 & \begin{bmatrix} \beta_{2j}WC - Y_1 \\ \varepsilon\beta_{2j}WC - Y_2 \end{bmatrix} & \mu_2\beta_{2j} \begin{bmatrix} WC \\ \varepsilon WC \end{bmatrix} \\ * & -S & 0 \\ * & * & -\mu_2R_a \end{bmatrix} < 0, \\ \begin{bmatrix} R & Y \\ * & Z \end{bmatrix} \geq 0,$$

where  $\beta_{2j}$  are defined by:

$$\begin{aligned} \beta_{11} &= e^{\alpha(\delta_1 - \mu_1)}, & \beta_{12} &= e^{\alpha(\delta_1 + \mu_1)}, \\ \beta_{21} &= e^{\alpha(\delta_2 - \mu_2)}, & \beta_{22} &= e^{\alpha(\delta_2 + \mu_2)}, \end{aligned} \quad (2.7)$$

and the matrices  $Y, Z$ , and  $\Psi_2$  are given by:

$$Y = [Y_1 \ Y_2], \quad Z = \begin{bmatrix} Z_1 & Z_2 \\ * & Z_3 \end{bmatrix}, \quad (2.8)$$

$$\begin{aligned} \Psi_2^{11} &= P^T(A_0 + \alpha I) + (A_0 + \alpha I)^T P + S \\ &\quad + \delta_2 Z_1 + Y_1 + Y_1^T, \\ \Psi_2^{12} &= P_1 - P + \varepsilon P^T(A_0 + \alpha I)^T + \delta_2 Z_2 + Y_2, \\ \Psi_2^{22} &= -\varepsilon(P + P^T) + \delta_2 Z_3 + 2\mu_2 R_a + \delta_2 R. \end{aligned}$$

Then, the gain:

$$L = (P^T)^{-1}W, \quad (2.9)$$

makes the error (2.6) of observer (2.5) exponentially convergent to the solution  $e(t)=0$ , with a decay rate  $\alpha$ .

In the following, the solution of the LMI problem corresponding to this theorem is written:

$$L = LMI_{obs}(\mu_2, \delta_2, \alpha) \quad (2.10)$$

### 2.2.5 Control Design

We first consider a controller  $u = Kx$ ,  $i = 1, 2$ , i.e., the ideal situation  $e(t) = 0$ ,  $x(t) = \hat{x}(t)$ , and:

$$\dot{x}(t) = Ax(t) + BKx(t - \delta_1(t)). \quad (2.11)$$

**Theorem 2.2.** [13] Suppose that, for some positive numbers  $\alpha$  and  $\varepsilon$ , there exists a positive definite matrix  $\bar{P}_1$ , matrices of size  $n \times n$ :  $\bar{P}$ ,  $\bar{U}$ ,  $\bar{Z}_1$ ,  $\bar{Z}_2$ ,  $\bar{Z}_3$ ,  $\bar{Y}_1$ ,  $\bar{Y}_2$  similar to (2.8) and an  $n \times m$  matrix  $W$ , such that the following LMI conditions hold:

$$\Gamma_{3i} = \begin{bmatrix} \Psi_3 & \begin{bmatrix} \beta_{1i}BW - \bar{Y}_1^T \\ \varepsilon\beta_{1i}BW - \bar{Y}_2^T \end{bmatrix} & \mu_1 \begin{bmatrix} \beta_{1i}BW \\ \varepsilon\beta_{1i}BW \end{bmatrix} \\ * & -\bar{S} & 0 \\ * & * & -\mu_1\bar{R}_a \end{bmatrix} < 0, \quad \forall i = 1, 2,$$

$$\begin{bmatrix} \bar{R} & \bar{Y}_1 & \bar{Y}_2 \\ * & \bar{Z}_1 & \bar{Z}_2 \\ * & * & \bar{Z}_3 \end{bmatrix} \geq 0,$$

where  $\beta_{1i}$ , for  $i = 1, 2$ , are defined by (2.7) and

$$\begin{aligned} \bar{\Psi}_3^{11} &= (A_0 + \alpha I)\bar{P} + \bar{P}^T(A_0 + \alpha I)^T + \bar{S} \\ &\quad + \delta_1\bar{Z}_1 + \bar{Y}_1 + \bar{Y}_1^T, \\ \bar{\Psi}_3^{12} &= \bar{P}_1 - \bar{P} + \varepsilon\bar{P}^T(A_0 + \alpha I)^T + \delta_1\bar{Z}_2 + \bar{Y}_2, \\ \bar{\Psi}_3^{22} &= -\varepsilon(\bar{P} + \bar{P}^T) + \delta_1\bar{Z}_3 + 2\mu_1\bar{R}_a + \delta_1\bar{R}. \end{aligned}$$

Then, the gain:

$$K = W\bar{P}^{-1}, \quad (2.12)$$

exponentially stabilizes the system (2.11) with the decay rate  $\alpha$  for all delay  $\delta_1(t)$ .

In the following, the solution of the LMI problem corresponding this theorem is written:

$$K = LMI_{con}(\mu_1, \delta_1, \alpha) \quad (2.13)$$

### 2.2.6 Global Stability of the Remote System

The gains  $K$  and  $L$  have to be computed in such a way they exponentially stabilize the global Master-Slave-Observer system despite the variable delays  $\delta_1(t)$  and  $\delta_2(t)$ . This global system is:

$$\begin{cases} \dot{x}(t) = Ax(t) + BKx(t - \delta_1(t)) + BKe(t - \delta_1(t)), \\ \dot{e}(t) = Ae(t) + LCe(t - \delta_2(t)). \end{cases} \quad (2.14)$$

A separation principle is then applied to the previous system. Then if  $L$  and  $K$  are chosen with respect to Sections 2.2.4 and 2.2.5, respectively, the global system is exponentially stable with the lowest decay rate.

## 2.3 Architecture of the Global Control System

### 2.3.1 Features of the Remote System

The remote system is based on a Master–Slave structure. To simplify the work of the Slave PC, the control and observation complexity is concentrated on the Master. For the main features of the system refer to Fig. 2.3. In the system, the robot Miabot of the company Merlin Systems Corp. Ltd., together with a PC, serves as the Slave. The Miabot works as an accessory device that communicates with the PC by the Bluetooth port and so we cannot use a buffer strategy directly on the robot. Because there is no time information included in the command of Miabot, we do not know when the control is applied. That means we cannot use a time stamp on Miabot. To simplify our problem and apply the theory of [13], we treat the time-delay of the Bluetooth between the PC and Miabot as a constant one. We add the delays  $d_1$ ,  $d_2$  into the respectively variable delays  $h_1(t)$ ,  $h_2(t)$ .

The transmission protocol UDP is applied to communicate the data between Master and Slave. To know the instant that the data was sent, time stamps are added to the data packets. The data structure of a list that serves as a buffer is introduced for the program to search for the data at the right instance. In all the lists, the control data are stored in a decreasing order of its transmission time. That is to say, the most recent sent data is always at the top of the list.

### 2.3.2 Synchronization of the Time Clocks

To synchronize the different time in the two PCs, we can add GPS to the system [13], but this increases the cost and is inflexible. Another way is to use a certain protocol such as NTP [20]. Because of different time clock times of the PCs, we have to make synchronize them from time to time. Our solution is to directly adapt the strategy of NTP in the program to calculate the time differences.

As shown in Fig. 2.4,  $k$  is the sequential number of the packets sent from the Master and  $k'$  is the number sent back.  $h_1(k)$  and  $h_2(k')$  refer to the respective delays

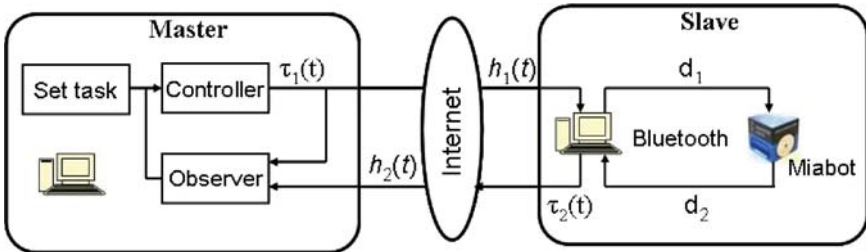
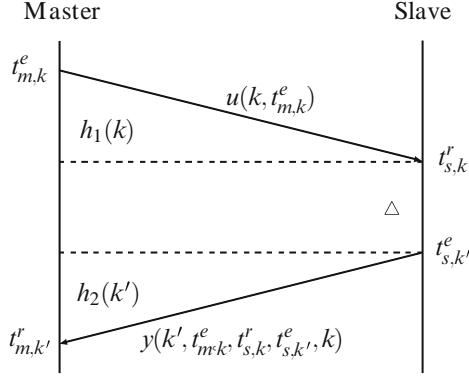


Fig. 2.3 Structure of the global system



**Fig. 2.4** Packets communication between the M/S

of the communication on Internet. To simplify the problem, we assume that  $h_1(k) = h_2(k')$ . If we define the time clock difference between the M/S as follows ( $t_s, t_m$  are the respectively time of the Slave and the Master):

$$\theta = t_m - t_s; \quad (2.15)$$

Then, we can calculate the time difference between the two PCs using the following equations ( $\Delta$  is the process time for Slave):

$$\theta(k, k') = (t_{m,k'}^r - t_{s,k'}^e + t_{m,k}^e - t_{s,k}^r)/2; \quad (2.16)$$

$$h_1(k) = h_2(k') = (t_{s,k}^r - t_{m,k}^e + t_{m,k'}^r - t_{s,k'}^e)/2; \quad (2.17)$$

That is to say, every time the Master receives a packet, the time clock difference between the M/S and the time delay of the Internet can both be measured. The values of  $\theta$  and  $h_m$  are contained in the control packets, so whenever the Slave receives a control, it can calculate the “target” time for applying the control.

### 2.3.3 Transmission and Receipt of The Control Data

The  $k^{\text{th}}$  data sent by the Master to Slave includes the control  $u(t_{m,k}^e)$  together with the time stamp when the packet is sent. At time  $t_{s,k}^r$ , when the Slave receives the data it can calculate the time delay because of the time stamp. If the delay equals  $h_{1m} - d_1$ , the Slave should immediately apply the command.

The control  $u$ , sent out by the Master at time  $t_{m,k}^e$ , is received by the Slave at time  $t_{s,k}^r > t_{m,k}^e - \theta$ . It will be inserted into the Slave input only at the predefined “target time”:  $t_{s,k}^{\text{target}} = t_{m,k}^{\text{target}} - \theta = t_{m,k}^e + h_{1m} - \theta$ . The corresponding waiting time  $h_{1m}$  is depicted in Fig. 2.5. This is realistic because the transmission delay is bounded by a known value  $h_{1m}$ . In this way, the Master knows the time when this control  $u(t_{m,k}^e)$  will be inserted into the Slave input.

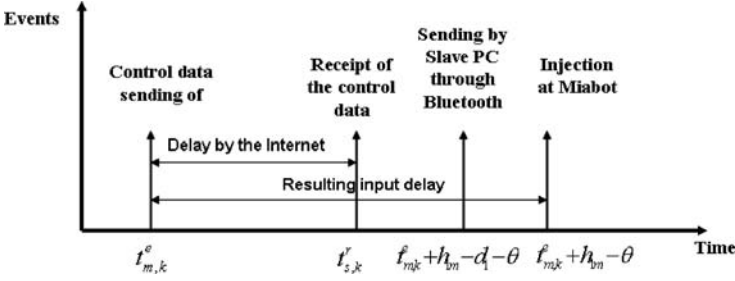


Fig. 2.5 Control data processing

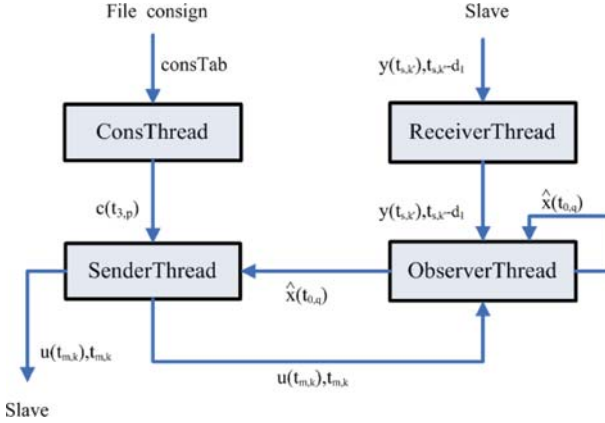


Fig. 2.6 Structure of the Master

### 2.3.4 The Structure of the Master

In order to implement the model for the remote control system, four-thread programs were designed to fulfill the functions of Controller and Observer in Fig. 2.3.

These four threads concurrently work as shown in Fig. 2.6. There are two buffers, *list\_U* and *list\_X*, which respectively keep the data sent out from the Master and the data of the estimated state of the Slave. The most recent calculated data is inserted at the beginning of the lists so that it is easier to find the right data we need and delete the obsolete ones.

Here we consider the following notation.  $p$  is the index of the different commands given by the ConsThread.  $k$  is the index of the control  $u$  given by the SenderThread.  $q$  is the index of estimation state given by the ObserverThread and the  $k'$  is the index of the measure given by the ReceiverThread.

- (a) ConsThread is a periodic thread that gets the tasks (it is the position where the user wants the motor to arrive) from a file given by the user. In this way, the user can freely change the task. The time period  $T_3$ , for this thread to continuously operate is set before the system begins to work. Its value should be greater than the response time of the Slave.

- (b) SenderThread is also a periodic thread that first gets the task ( $c(t_{3,p})$ ) designed by the user. Considering the mechanical character of the Miabot, we choose 0.1 s as the time period. Then, it calculates the control data to send to the Slave. The most recent  $\hat{x}(t_{0,q})$  can be found at the beginning of the *list\_X*; then, the command data, together with the system time, is sent to the slave through a socket. While, at the same time, it is inserted into the *list\_U* for the use by the ObserverThread.
- (c) ReceiverThread is an event-driven thread. As data arrives from the slave, it first checks whether there are any packets lost. The maximum number of packets that can be lost without implying instability of the system is an open issue in our research. Then, according to the time stamp, the time clock difference and the time-delay are calculated; meanwhile, the most recent data is sent to the thread of the ObserverThread.
- (d) ObserverThread is the most important part of the program. It mainly serves as the Observer in the system model. The main task is to estimate the *present* position and speed of the motor. To estimate the continuous state of the Miabot, the time period of this thread should be small enough. In our program, we choose 0.01 s. As we can see from the result of the experiment (Sect. 2.3.6), this value is suitable. To accomplish the function of the Observer, it must find out the command  $u$  which has been applied to the slave system and the estimated motor position at the time when the information is sent from the slave.

As it is illustrated in Fig. 2.7, in order to determine  $\hat{y}(t_{2,k'})$ , it is necessary to find, in the *list\_X*, the closest state estimation  $\hat{x}$  with regard to the date  $t_{s,k'} - d_1$ . And we can get the control data  $u$  in the *list\_U* with the time stamp of time  $h_{1m}$  before. So, according to the equation (2.4), the estimated state can be obtained. As we can see from Fig. 2.7, in order to find the state  $\hat{x}(t_{0,q})$  at the time nearest to  $t_{s,k'} - d_1$ , the time period of this thread should be small enough.

### 2.3.5 The Structure of the Slave

The Slave does not need power computation abilities, because it just needs to communicate with the Master and the Miabot. As we can see from Fig. 2.8, this program is divided into two threads: ReceiveThread and SendThread. As we need to apply the control data with the time delay of  $h_{1m}$  after the time stamp, a *list\_Y* is used to contain the control data temporarily, in which all the nodes are sorted in the order

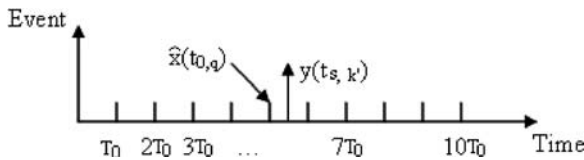
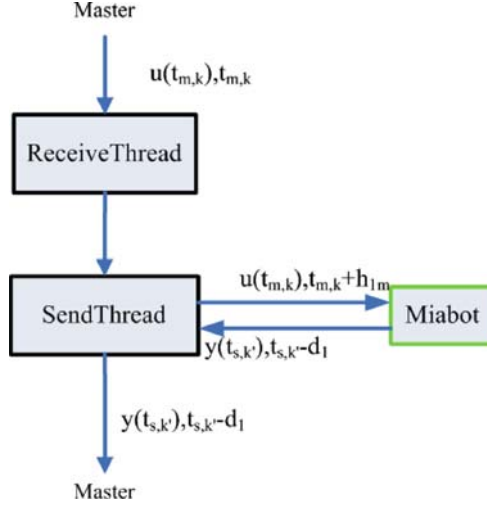


Fig. 2.7 Packet Sequences



**Fig. 2.8** Structure of the Slave

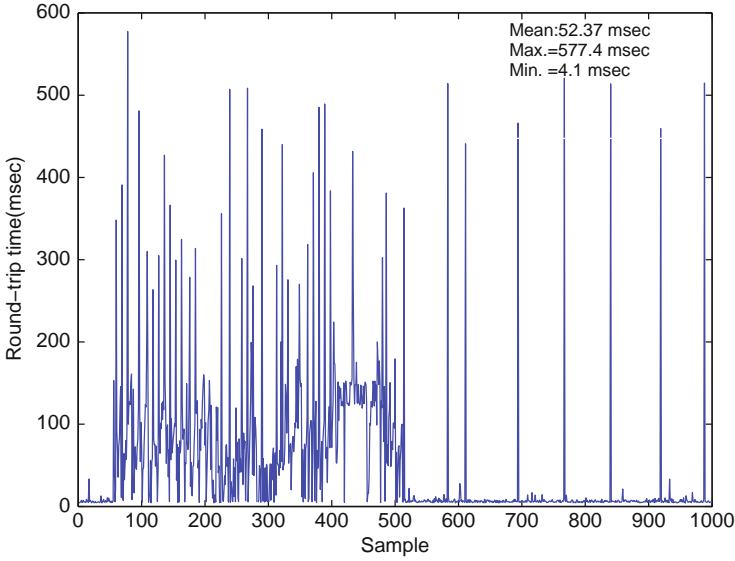
according to the time stamps. That means the most recent control is inserted at the end of the list.

- (a) ReceiveThread is an event-driven thread that is activated by the control data arriving from the Master. The control data is inserted into the proper position in the list *list\_Y* according to its time stamp. If the time stamp falls before the oldest data in the list, the packets are out of order throughout the Internet and the data is discarded. If there are several packets lost, the stability of the system is not affected, since the Master's SendThread has a sufficiently high transmission frequency.
- (b) SendThread is used to apply the control to the Miabot as well as to get its real position, and send the data back to the Master. First, the thread takes the packet at the beginning of *list\_Y*. Thanks to the value of the difference time clock between Master and Slave in the packet, we can calculate the “target” time to apply the control. Then the control is sent to the Miabot by the Bluetooth port at the right time, when the Miabot sends back the measure  $y(t_{s,k'}^e)$  of its position. This value is sent to the Master with the time stamp  $t_{s,k'} - d_1$  where  $t_{s,k'}$  is the reception time by the Slave PC.

### 2.3.6 Experimental Study

After identification of the Miabot, we get the following model:

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & -10 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0.014 \end{bmatrix} u(t - \delta_1(t)) \\ y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t). \end{cases} \quad (2.18)$$



**Fig. 2.9** The RTT between the two PCs by Internet (40 km away)

We have continuously tested the RTT (Round-trip-time) between the two PCs by the ICMP (Internet Control Message protocol) as showed in Fig. 2.9. From these tests, considering also the Bluetooth transmission delays and the sampling delays, we take the value  $\delta_1 = \delta_2 = 0.4$  s and  $\mu_1 = \mu_2 = 0.1$  s. If there any times at which the time-delays are greater than 0.5 s, we treat the packets as being lost. The gains  $K$  and  $L$  have to be computed in such a way that they exponentially stabilize the global Master-Slave-Observer system despite the variable delays  $\delta_1(t)$  and  $\delta_2(t)$ . According to [13], we get  $\alpha = 0.96$  when the gain  $L$  is chosen as:

$$L = \begin{bmatrix} -1.4 \\ -0.13 \end{bmatrix}. \quad (2.19)$$

The gain  $K$  is as follows:

$$K = \begin{bmatrix} -702 & -70 \end{bmatrix} \quad (2.20)$$

The experiment is done on two computers separated about 40 k away. The Master program runs on the remote computer with an advanced computing capability, the slave program on the local one. We get the result shown in the Fig. 2.10, in which the dash-dot line represents the set values; the solid lines represent the robot's estimated state, the position and the speed; the dotted line correspond to the real position. Fig. 2.11 represents the sampled control data sent to the Slave.

Note that all the data in the figure are obtained from the Master, so the data of the real position of the Miabot (dotted line) lags behind the estimated one. This illustrates the fact that, despite the time delays of the Internet and the Bluetooth, the Master predicts an estimate of the Slave's state.

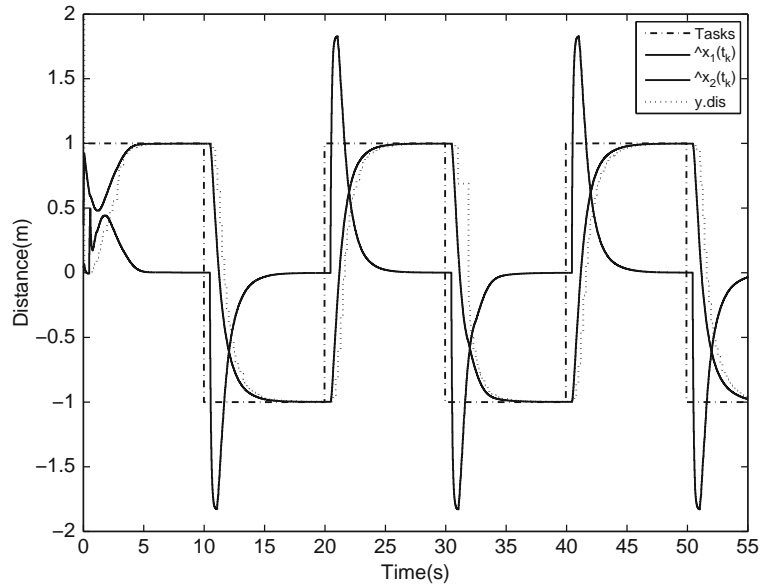


Fig. 2.10 Results of remote experiment

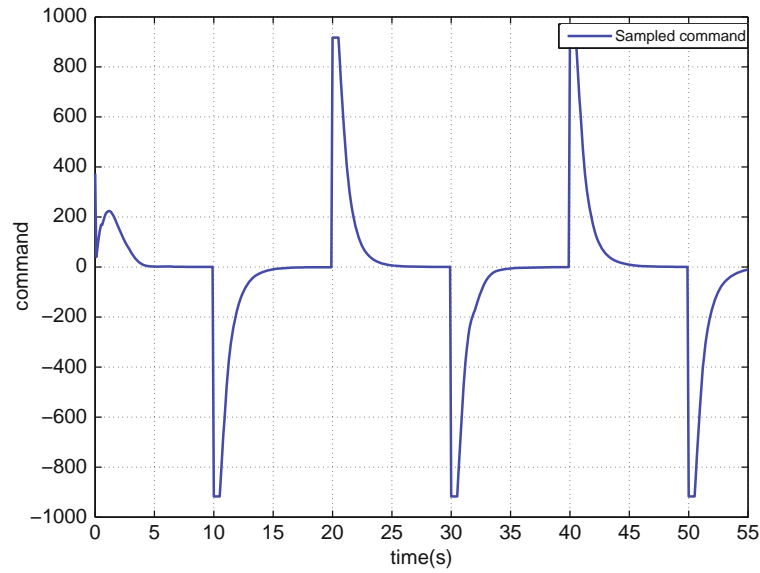


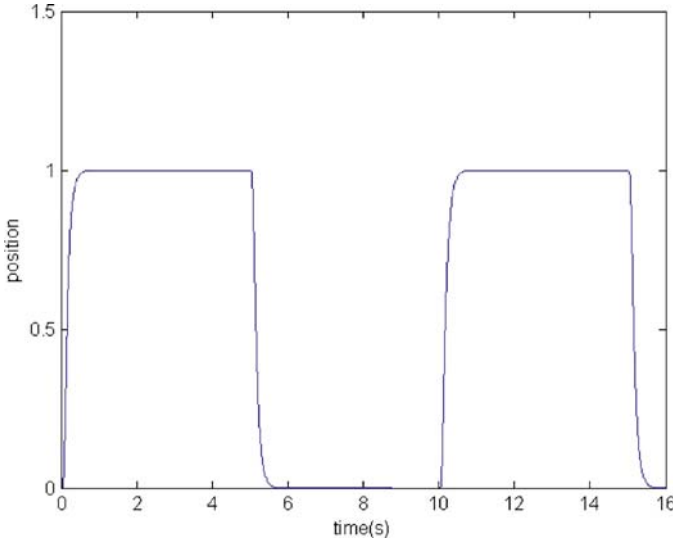
Fig. 2.11 The corresponding Slave control of centralization experiment

## 2.4 Performance Enhancement by a Gain Scheduling Strategy

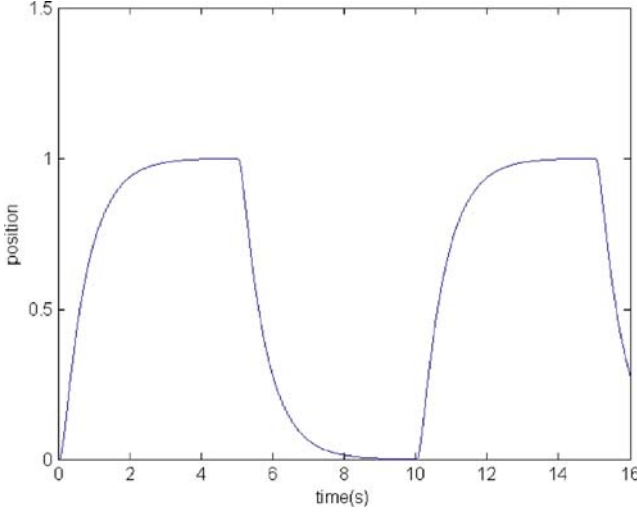
### 2.4.1 Effects of Time-Delay on the Performance and the System Stability

The time-delay of the Internet varies greatly between the rush hour and the idle time period (Fig. 2.9). To guarantee the exponential stabilization, we have to choose the maximum time-delay, whereas most of the time, the time-delay is much smaller. In other words, the performance is decreased. Comparing the two simulation results from Matlab, Figs. 2.12 and 2.13, we can see that when the  $h_m$  is smaller, the value of  $\alpha$  is larger and the task is quickly accomplished. In the table in Fig. 2.14, we list several corresponding values of  $h_m$  and  $\alpha$ . It is clearly that increasing  $h_m$  means decreasing the performance of the system.

To enhance the performance and make the system adaptable to the changeable time-delay of the Internet, we have designed two controllers corresponding with two bounds of time-delay. The controller switches on the time-delay function. The switching signal is given by  $\sigma(t) = \gamma(t - \xi)$ , where  $\xi$  is the time-delay of the signal due to the Internet and calculation. In order to guarantee the *uniform* exponential stability, our solution is to find a common Lyapunov function for both closed loops [23]. Of course, for greater delay values, the performance cannot be guaranteed anymore and an alternative solution has to be considered. In our system, we give a command for the robot to stop until the communication comes back to an adequate level.



**Fig. 2.12**  $h_m = 0.05\text{ s}$ ,  $\alpha = 8.74$



**Fig. 2.13**  $h_m = 0.5\text{s}$ ,  $\alpha = 0.96$

$h_m$	0.02	0.04	0.06	0.08	0.1	0.2	0.3	0.4	0.5
$\alpha$	9.99	9.99	7.54	5.94	4.91	2.59	1.71	1.25	0.96

**Fig. 2.14** The relation between  $h_m$  (s) and  $\alpha$

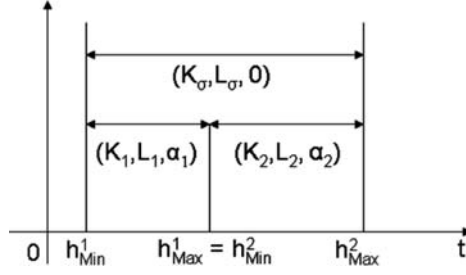
### 2.4.2 Uniform Stability with Gain Scheduling

In order to reach a larger value for the exponential convergence, we propose both switching controller and observer gains. The switching signals  $\sigma_1(t)$  and  $\sigma_2(t)$  chosen are functions of the time delays  $\delta_1(t)$  and  $\delta_2(t)$ . For the sake of simplicity, they can only take two values:

$$\sigma_i(t) = j, \quad \text{if } \delta_i(t) \in [h_{\text{Min}}^{ij}, h_{\text{Max}}^{ij}], \quad i, j = 1, 2 \quad (2.21)$$

Considering every time-delay zone, we have to compute the gains  $K_1$ ,  $K_2$  and  $L_1$ ,  $L_2$  in such a way that they exponentially stabilize the global Master–Slave–Observer system despite the variable delays  $\delta_1(t)$  and  $\delta_2(t)$ . This global system is:

$$\begin{cases} \dot{x}(t) = Ax(t) + BK_{\sigma_1(t)}x(t - \delta_1(t)) + BK_{\sigma_1(t)}e(t - \delta_1(t)), \\ \dot{e}(t) = Ae(t) + L_{\sigma_2(t)}Ce(t - \delta_2(t)), \\ \sigma_1(t) = \gamma_1(t - \delta_1(t) - \delta_2(t)), \\ \sigma_2(t) = \gamma_2(t - \delta_2(t)). \end{cases} \quad (2.22)$$



**Fig. 2.15** Uniform stability of the system

$\gamma_i(t)$  are the detection functions of  $\delta_i(t)$ . These functions are delayed because the Master needs to receive the last packet to calculate the delays. Therefore each gain is activated for a certain period  $(\delta_1(t) + \delta_2(t))$  or  $\delta_2(t))$ .

Each gain is computed using

$$K_j = \text{LMI}_{\text{con}}((h_{\text{Max}}^{1j} - h_{\text{Min}}^{1j})/2, (h_{\text{Max}}^{1j} + h_{\text{Min}}^{1j})/2, \alpha_{\text{con}})$$

$$L_j = \text{LMI}_{\text{obs}}((h_{\text{Max}}^{2j} - h_{\text{Min}}^{2j})/2, (h_{\text{Max}}^{2j} + h_{\text{Min}}^{2j})/2, \alpha_{\text{obs}}).$$

A sufficient condition to prove the uniform stability of the switching closed loop is to find a common Lyapunov-Krasovskii functional for all gains. This functional has to take into account all admissible delays, i.e.,  $\forall \delta_i(t) \in [h_{\text{Min}}^{i1}, h_{\text{Max}}^{i2}]$ . For each gain, Fig. 2.15 shows the regions where the exponential stability occurs.

The following theorems give sufficient conditions to prove the uniform stability of the switching closed loop.

**Theorem 2.3.** Suppose that, for a given switching observer gains  $L_{\sigma_2(t)}$ , for some positive scalars  $\alpha$  and  $\varepsilon$ , there exists  $n \times n$  matrices  $0 < P_1, P, S, Y_1, Y_2, Z_1, Z_2, Z_3, R, R_a$  such that the LMI conditions 2.1 and the following ones hold for  $i = 1, 2, j = 1, 2$

$$\begin{bmatrix} \Psi_2 & \begin{bmatrix} \beta_{2j} P^T L_i C - Y_1 \\ \varepsilon \beta_{2j} P^T L_i C - Y_2 \end{bmatrix} & \mu_2 \beta_{2j} \begin{bmatrix} P^T L_i C \\ \varepsilon P^T L_i C \end{bmatrix} \\ * & -S & 0 \\ * & * & -\mu_2 R_a \end{bmatrix} < 0,$$

where the matrices  $\Psi_2$  and  $\beta_{2j}$  are the same as in Theorem 2.1.

Then the error of observer exponentially converge to the solution  $e(t) = 0$ , with a decay rate  $\alpha$ .

*Proof.* Consider the Lyapunov-Krasovskii functional (2.3). Following the same proof as in [13], one gets the following sufficient conditions for  $j = 1, 2$ :

$$\begin{bmatrix} \Psi_2 & \begin{bmatrix} \beta_{2j} P^T L_{\sigma_2(t)} C - Y_1 \\ \varepsilon \beta_{2j} P^T L_{\sigma_2(t)} C - Y_2 \end{bmatrix} & \mu_2 \beta_{2j} \begin{bmatrix} P^T L_{\sigma_2(t)} C \\ \varepsilon P^T L_{\sigma_2(t)} C \end{bmatrix} \\ * & -S & 0 \\ * & * & -\mu_2 R_a \end{bmatrix} < 0,$$

Then by convexity, one obtains the conditions of the previous theorem.  $\square$

**Theorem 2.4.** Suppose that, for a given switching state feedback  $K_{\sigma_1(t)}$ , for some positive numbers  $\alpha$  and  $\varepsilon$ , there exists a positive definite matrix  $\bar{P}_1$ , matrices of size  $n \times n$ :  $\bar{P}$ ,  $\bar{U}$ ,  $\bar{Z}_1$ ,  $\bar{Z}_2$ ,  $\bar{Z}_3$ ,  $\bar{Y}_1$ ,  $\bar{Y}_2$  similar to (2.8), such that LMI conditions 2.1 and the following ones hold for  $i = 1, 2$ ,  $j = 1, 2$

$$\begin{bmatrix} \Psi_3 & \begin{bmatrix} \beta_{1i}BK_j\bar{P} - \bar{Y}_1^T \\ \varepsilon\beta_{1i}BK_j\bar{P} - \bar{Y}_2^T \end{bmatrix} & \mu_1 \begin{bmatrix} \beta_{1i}BK_j\bar{P} \\ \varepsilon\beta_{1i}BK_j\bar{P} \end{bmatrix} \\ * & -\bar{S} & 0 \\ * & * & -\mu_1\bar{R}_a \end{bmatrix} < 0,$$

where the matrices  $\Psi_3$  and  $\beta_{1j}$  are the same as in Theorem 1. Then the closed loop is exponentially stable with the decay rate  $\alpha$  for all delay  $\delta_1(t)$ .

*Proof.* Proof: Same as in the observer case.  $\square$

### 2.4.3 Gain Scheduling Experiments

Considering the initial approach, i.e., without switching gains, the maximum exponential convergence is obtained at  $\alpha_{\text{control}} = \alpha_{\text{observer}} = 0.96$ .

Consider two zones of delay with  $\delta_1^1 = \delta_2^1 = 0.04$  s,  $\mu_1^1 = \mu_2^1 = 0.04$  s, and  $\delta_1^2 = \delta_2^2 = 0.29$  s,  $\mu_1^2 = \mu_2^2 = 0.21$  s. This means that the gains switch, when the delay crosses the value of 0.08 s. According to Theorems 2.3 and 2.4, the maximum exponential convergence ensuring the global stability occurs when:  $\alpha_{\text{control}1} = 2.1$ ,  $\alpha_{\text{observer}1} = 2.2$ , and  $\alpha_{\text{control}2} = \alpha_{\text{observer}2} = 1$ .

Note that because the global stability is checked after the computation of the gains, these values are not optimal. To get optimal value, needs to find the control/observer gains ensuring the exponential convergence and the global stability at the same time with a LMI problem.

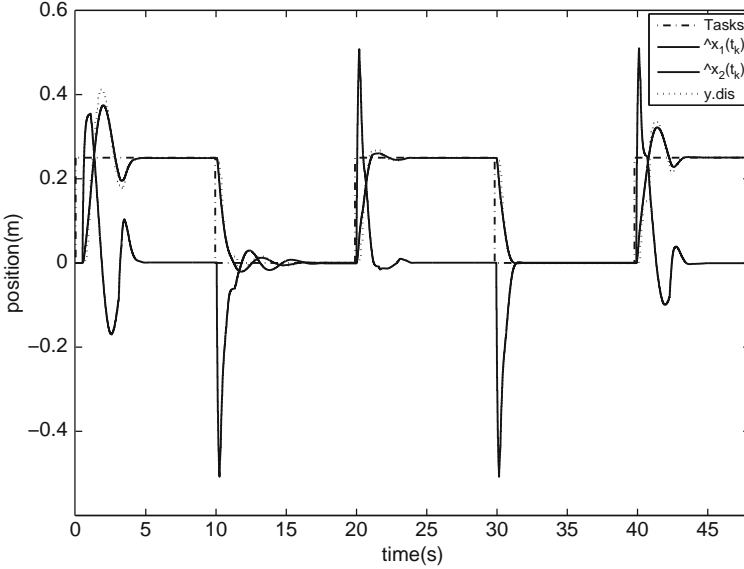
The gains  $K_i$  and  $L_i$  ( $i = 1, 2$ ) are:

$$L_1 = \begin{bmatrix} -3.01 \\ -0.77 \end{bmatrix}, \quad K_1 = [-1, 659 \quad -260].$$

$$L_2 = \begin{bmatrix} -1.4 \\ -0.16 \end{bmatrix}, \quad K_2 = [-1, 015 \quad -100].$$

### 2.4.4 Result of Remote Experiment

The experiment is done in the same situation as mentioned in the preceding sections. The result is shown in Fig. 2.16, in which the dash-dot line represents the set values; the solid lines represent respectively the robot's estimated position and speed;



**Fig. 2.16** Results of remote experiment

the dotted line corresponds to the real position of the Miabot. Fig. 2.17 is the corresponding variable time-delays, which comprises the time-delay of sampling and communication of the Bluetooth (we consider it as constant time-delay, here we take the value of 40 ms). In Fig. 2.18, the solid line represents the sampled control sent to Slave, and the dash-dot line and dotted one represent the command for the zone one and two respectively. Fig. 2.19 shows the time point when the system switches enter the values for the two zones.

Because the maximal speed of the Miabot is 3.5 m/s [24], the corresponding command value is 2,000 (in open loop). But to guarantee the linear character of the Miabot, we select the command value not greater than 1,000 and the speed 1.7 m/s. The controller gains are those of the last section. Despite their high value, one can notice that the control signal (Fig. 2.18) does not exceed the limit value as well as the speed (Fig. 2.16 solid line) that validates the linearity assumption.

On Fig. 2.16, one can notice three kinds of step response. The first one corresponds to the case where the control switches often during the response. In that case, only the global stability is guaranteed. During the second step, only the second zone is active, i.e., only the gains  $K_2$  and  $L_2$  are active ( $\alpha = 1$ ). In this case, some performance is guaranteed. In the last kind of response, only the first zone is active because the delays are small. In that case, the performance is better ( $\alpha = 2.1$ ): The response time is smaller and the damping is greater.

As it is clearly shown in Fig. 2.16, the global stability of the closed loop is maintained despite the fact that some assumptions are not satisfied. With respect to the Bluetooth assumption, it was considered constant, whereas in reality it varies (the minimum delay recorded as less than 4, ms). And with respect to synchronization, symmetric delays were needed and in the experiment it was clearly not the case.

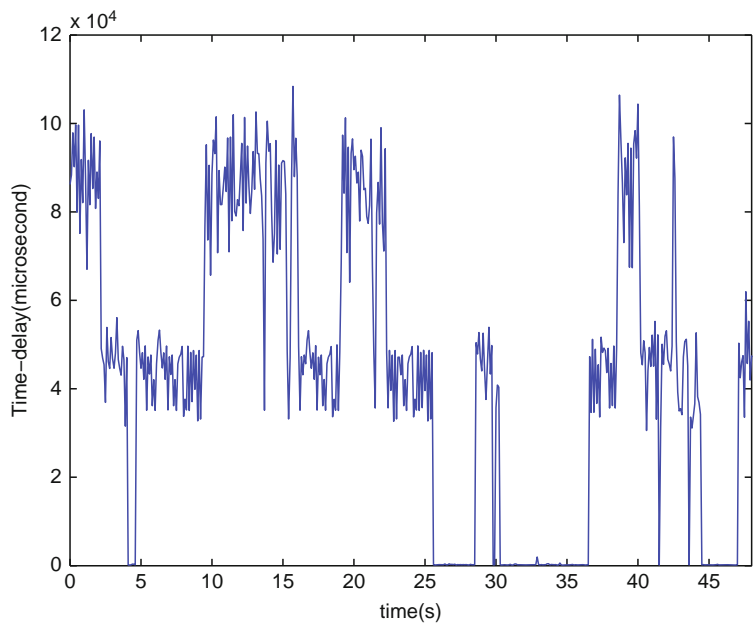


Fig. 2.17 The corresponding variable time-delays

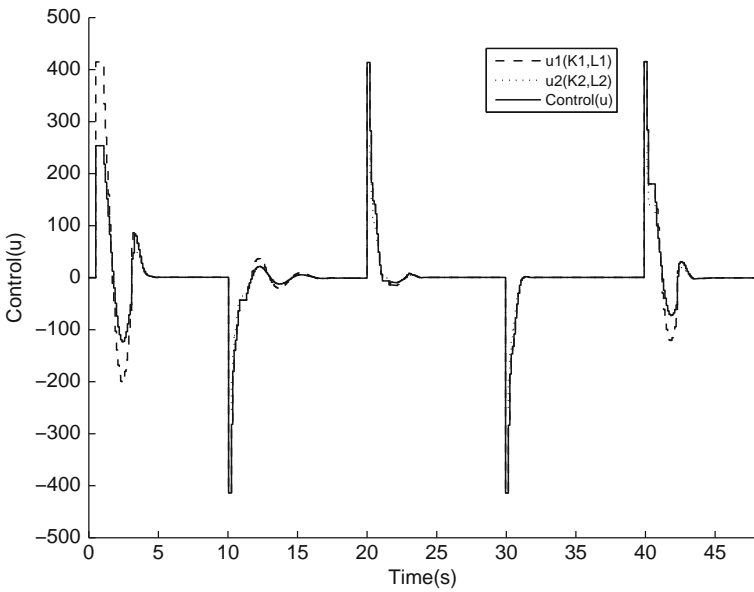


Fig. 2.18 The corresponding Slave control

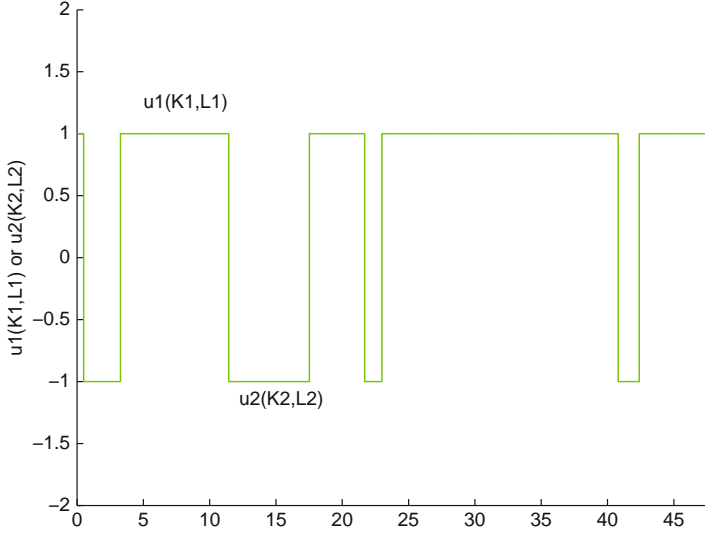


Fig. 2.19 The switching plan

## 2.5 Conclusion

In addition to some fundamental results, an experimental platform has been developed to illustrate the results of the network-based control theory. This platform is able to control a slave through a network and joins skills in automatic control, computer science, and networks.

The experimental results confirm the theory: (1) The exponential stability is obtained in both the time-delay zones and *uniform* stability is guaranteed (2) The experimental performances are shown to be better when considering two zones of time delay instead of one.

Considering the variation of time-delays, more than two-zone-switching signals can be selected in order to enhance the performance of the global system. The LMI conditions, in that case, have an increased size and are straightforwardly inspired from Theorems 2.3 and 2.4 but are not investigated here.

A way to improve the presented results is to propose a “one shot algorithm” that allows finding the optimal gains in terms of exponential convergence. Another trend is to investigate a solution without the input buffer. Without the buffer, the input delay will be smaller, ensuring better performance and the slave will need less memory to run [25].

A last perspective is to consider the improvement of the network communication by, for example, developing dedicated protocols that minimize the time delays and enhance the clock synchronization.

## References

1. Yu M., Wang L., and Chu T., "An LMI approach to network control systems with data packet dropout and transmission delays," *MTNS '04 Proceedings of Mathematical Theory Networks and Systems*, Leuven, Belgium, 2004.
2. Richard J.P. and Divoux T., *Systèmes commandés en réseau*. Hermes-Lavoisier, IC2, Systèmes Automatisés, 2007.
3. Georges J.-P., Divoux T., and Rondeau E., "Confronting the performances of a switched ethernet network with industrial constraints by using the network calculus," *International Journal of Communication Systems (IJCS)*, vol. 18, no. 9, pp. 877–903, 2005.
4. Chiasson J. and Loiseau J.J., *Applications of time delay systems*. Springer, New York, vol. 352, 2007.
5. Hale J. and Lunel S., *Introduction to functional differential equations*, Springer, New York, 1993.
6. Kolmanovskii V. and Myshkis A., *Applied theory of functional differential equations*. Dordrecht: Kluwer, 1999.
7. Niculescu S.-I., *Delay effects on stability: A robust control approach*. Springer, New York, vol. 269, 2007.
8. Richard J.P., "Time delay systems: an overview of some recent advances and open problems," *Automatica*, vol. 39, pp. 1667–1694, 2003.
9. Almutairi N.B., Chow M.Y., and Tipsuwan Y., "Network-based controlled dc motor with fuzzy compensation," *Proceedings of the Annual Conference on Industrial Electronics Society*, Denver, pp. 1844–1849, 2001.
10. Huang J.Q. and Lewis F.L., "Neural-network predictive control for nonlinear dynamic systems with time delays," *IEEE Transactions on Neural Networks*, vol. 14, no. 2, pp. 377–389, 2003.
11. Leleve A., Fraisse P., and Dauchez P., "Telerobotics over IP networks: Towards a low-level real-time architecture," *ROS'01 International Conference on Intelligent Robots and Systems*, Maui, Hawaii, October 2001.
12. Witrant E., Canudas-De-Wit C., and Georges D., "Remote stabilization via communication networks with a distributed control law," *IEEE Transactions on Automatic Control*, 2007.
13. Seuret A., Michaut F., Richard J.P., and Divoux T., "Networked control using gps synchronization," *Proceedings of ACC06, American Control Conference*, Mineapolis, USA, June, 2006.
14. Sandoval-Rodriguez R., Abdallah C., Jerez H., Lopez-Hurtado I., Martinez-Palafox O., and Lee D., *Networked Control Systems: Algorithms and Experiments, A Applications of time delay systems*, vol. 352. Springer, New York, pp. 37–56, 2007.
15. Azorin J.M., Reinoso O., Sabater J.M., Neco R.P., and Aracil R., "Dynamic analysis for a teleoperation system with time delay," *Conference on Control Applications*, June 2003.
16. Fattouh A. and Sename O., " $H^\infty$ -based impedance control of teleoperation systems with time delay," *Fourth Workshop on Time Delay Systems*, September 2003.
17. Niemeyer G. and Slotine J.-J., "Towards force-reflecting teleoperation over the internet," *IEEE International Conference on Robotics & Automation*, 1998.
18. Chen Z., Liu L., and Yin X., "Networked control system with network time-delay compensation," *Industry Applications Conference, Fourteen IAS Annual Meeting*, vol. 4, pp. 2435–2440, 10, 2005.
19. Seuret A., Dambrine M., and Richard J.P., "Robust exponential stabilization for systems with time-varying delays," *Proceedings of TDS04, Fifth IFAC Workshop on Time Delay Systems*, Leuven, Belgium, September 2004.
20. Mills D.L., "Improved algorithms for synchronizing computer network clocks," *IEEE/ACM Transactions On Networking*, vol. 3, no. 3, pp. 245–254, June 1995.
21. Seuret A., Fridman E., and Richard J.P., "Sampled-data exponential stabilization of neutral systems with input and state delays," *Proceedings of IEEE MED 2005, Thirteen Mediterranean Conference on Control and Automation*, Cyprus, 2005.
22. Fridman E., Seuret A., and Richard J.P., "Robust sampled-data stabilization of linear systems: An input delay approach," *Automatica*, vol. 40 no. 8 19, pp. 1441–1446, 2004.

23. Liberzon D., *Switching in Systems and Control*, T. Basar, Ed. Birkhäuser, Briton, FL, 2003.
24. Merlin Systems Corporation Ltd., *Miabot PRO BT v2 User Manual*, rev. 1.3 ed., 2004.
25. Seuret A. and Richard J.P., "Control of a remote system over network including delays and packet dropout," *IFAC World Congress*, Seoul, Korea, July, 2008.

## Chapter 3

# Developments in Control of Time-Delay Systems for Automotive Powertrain Applications

Mrdjan Jankovic and Ilya Kolmanovsky

**Abstract** In this chapter, the authors provide an overview of several application problems in the area of automotive powertrain control, which can greatly benefit from applications of analysis and control design methods developed for time-delay systems. The first two applications considered concern, respectively, the idle speed control (ISC) and air-to-fuel ratio control in gasoline engines. The third application concerns an estimation problem in modern diesel engines with exhaust gas recirculation and variable geometry turbocharging. The nature of the delays and the role played by them in these applications is highlighted and the imposed performance limitations are discussed. Links are provided with theoretical literature on the analysis and control of time-delay systems and modeling details are discussed to a level that will permit other researchers to use the associated models for simulation case studies.

**Keywords:** Air-to-fuel ratio control · Automotive powertrain control · Linear matrix inequality · Lyapunov function · Observer design · Idle speed control

### 3.1 Introduction

In this chapter, the authors focus on application problems in the area of automotive powertrain control, where the issues related to the treatment of time-delays constitute important design considerations. New engines and transmissions are introduced every year into production, and a broad range of advanced technologies are being continuously developed for future applications. Powertrain control is playing an increasingly important role in the automotive development cycle, and is a key enabler for future improvements in fuel economy and emissions. Since delays are common in powertrain systems, effective techniques for controlling time-delay systems can have a substantial impact on these applications.

Here, three powertrain control applications with delays are discussed. The first two applications concern, respectively, the idle speed control (ISC) and air-to-fuel ratio control in gasoline engines. The third application concerns an estimation problem in modern diesel engines with exhaust gas recirculation, and variable geometry turbocharging. In all of the applications, the authors' objective will be to highlight the character and the role the delay is playing and discuss performance limitations that it imposes. We will also provide links with theoretical literature on analysis and control of time-delay systems and discuss modeling details to a level that will permit others to use the associated models for the simulation case studies. The linear matrix inequality (LMI) approach used in the first two chapters will also be used here.

In the ISC case, the delay arises because of finite time between intake and power strokes of the engine and is generally assumed to be half of the engine cycle. The fueling rate is determined according to the predicted airflow at the time of the intake stroke [39] and this prediction can be made so accurate that the torque can be assumed to be proportional to the airflow delayed only by the intake to power delay but unaffected by the delays in the fueling system. In the air-to-fuel ratio control problem, the delays in the fueling system must be considered explicitly as this loop compensates for the air-to-fuel ratio disturbances. Finally, in the diesel engine, the delay is due to finite time between intake and exhaust strokes of the engine and can be assumed to be about three-fourth of the engine cycle.

### 3.2 Idle Speed Control

ISC is one of the key control functionalities of modern gasoline and diesel engines [19]. It maintains engine running when the driver foot is off the gas pedal.

A schematic block diagram of the ISC closed-loop system for a gasoline engine is presented in Fig. 3.1. The position of the electronic throttle and the spark timing

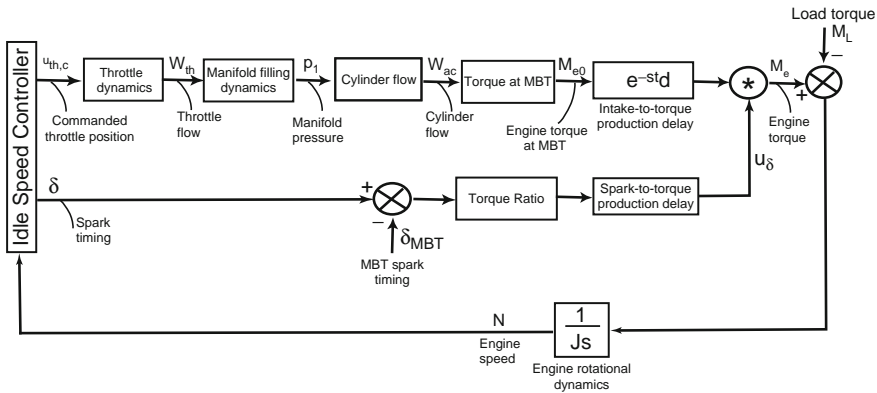


Fig. 3.1 Block diagram for ISC

are control inputs and the engine speed is a controlled output. The engine speed is assumed to be measured (in reality, it is fairly accurately estimated from the crankshaft position sensor signal). The objective of ISC is to maintain engine speed at a given set-point (around 625 rpm in drive gear and 650 rpm in neutral gear during fully warm operation) despite measured and unmeasured torque disturbances due to power steering, air-conditioning turn on and off, transmission engagement, alternator load changes, and the like. Both throttle and spark have an effect on the engine torque and can compensate the effects of disturbances. Tight performance requirements are imposed to contain the engine speed dip (maximum engine speed decrease when disturbance hits) within 80 rpm and engine speed flare (maximum engine speed increase when disturbance hits) within 120 rpm.

Increasing engine throttle opening causes air flow through the throttle to increase. The increase in the throttle flow causes the intake manifold pressure to increase, which in turn increases the airflow into the engine cylinders. Larger engine torque is produced as the cylinder air flow increase is proportionally matched by the increase in the engine fueling rate. Adjusting the spark timing shifts the start of combustion relative to the revolution of the crankshaft. The spark timing (in degree of crankshaft revolution) at which the maximum torque is produced is referred to as maximum brake torque (MBT) spark timing. Typically, the spark timing can only be retarded relative to the MBT spark timing, which causes engine torque to decrease. The maximum feasible spark retard is limited by combustion stability constraints. To ensure that spark has a bidirectional authority over the engine torque, the set-point for spark-timing in steady-state is retarded from MBT. This steady-state spark retard is referred to as *spark reserve*.

The need to maintain spark reserve results in fuel consumption increase as compared with the case when spark timing is maintained at MBT. As we will see shortly, the delay in this system is a key factor that prevents making throttle to engine speed control loop faster and eliminating the spark reserve. Another opportunity to reduce fuel consumption during idling is to lower idle speed set-point, provided this can be sustained by vehicle accessories. Lowering idle speed set-point makes the problem harder in two ways. First, the delay, which, as we will see shortly, is inversely proportional to engine speed, becomes larger. Second, the engine speed excursions, especially dips, must be much tighter controlled as engine speed decrease below the so-called “fishhook” point results in a rapid increase in engine friction (due to changing properties of the lubricant) and can easily produce an engine stall. Engine stall has a negative customer impact and the number of engine stalls is one of the key quality indicators (TGW—things gone wrong).

Historically, ISC is related to one of the oldest closed-loop systems discussed in the controls literature, the so called Watt’s governor (1787), which may be viewed as a speed controller for a steam engine. In older gasoline engines with a mechanical throttle (i.e., throttle mechanically connected to driver gas pedal), a dedicated actuator called air-by-pass valve was used for ISC [20] but in modern engines the introduction of an electronic throttle, decoupled from the driver’s gas pedal, has permitted to eliminate the air-by-pass valve. This did not make the control problem easier as electronic throttle designed for regular driving operates only in a narrow

range during idle where small imperfections may have large impact. Robustness to changes due to aging and manufacturing tolerances, operation across a wide range of environmental conditions (ambient temperature, pressure, humidity, etc.) and constant pressures to lower the cost of the hardware components (and achieve same quality of control with less capable components) are some of other sources of continuing challenges for ISC.

For these reasons, ISC still remains a relevant case study for application of advanced control methods despite numerous published controls approaches (see, e.g., [20] for a survey). In particular, it is clear from the above discussion that more effective techniques for controlling time-delay systems can have a substantial impact in improving fuel consumption during idling.

In what follows, we will first discuss a simple model suitable for initial studies of ISC control designs and the effects of the delay. We will then comment on some of the practical and advanced approaches proposed for dealing with this control problem in the literature.

### 3.2.1 A Model for ISC

A simplified model, based on the work of [28], is now discussed to illustrate the delay-related aspects of ISC. The model is a continuous-time, mean-value model in which cyclic fluctuations of engine operating variables are replaced by their cycle averages. In addition, several approximations are made that are valid locally around idle operating point. With these approximations, the engine speed dynamics have the form

$$\dot{N} = \frac{1}{J/(30/\pi)} (M_e - M_L), \quad (3.1)$$

where  $N$  is the engine speed in rpm,  $J$  is the engine inertia in  $\text{kg m}^2$ ,  $(30/\pi)$  is a conversion factor from rad/s to rpm,  $M_e$  is the torque produced by the engine (Nm), and  $M_L$  is the load torque in Nm. The engine torque is represented as a product of engine torque at MBT,  $M_{e0}$ , and torque sensitivity to spark (also referred to as the torque ratio),  $u_\delta$ , which is a function of spark retard:

$$M_e(t) = M_{e0}(t - t_d)u_\delta(t). \quad (3.2)$$

Here  $t_d$  is the delay between the intake stroke of the engine and torque production. Hereafter, we assume that we can manipulate  $u_\delta$  as if it was a control input and we neglect the delay between spark timing and torque production, noting that omitting the spark to torque delay, while reasonable, is not uniformly agreed, see e.g., [15]. The engine torque at MBT spark timing,  $M_{e0}$ , is a function of cylinder flow,  $W_{ac}$  (kg/s), and engine speed,  $N$ :

$$M_{e0} = \frac{k_1}{N} W_{ac}, \quad (3.3)$$

where  $k_1$  is a parameter. The cylinder flow is a function of the intake manifold pressure,  $p_1$  (kPa) and engine speed,  $N$ ,

$$W_{ac} = \frac{k_2}{k_1} p_1 N + k_0, \quad (3.4)$$

and  $k_0, k_2$  are constant parameters. The intake manifold filling dynamics are based on the ideal gas law and constant gas temperature (isothermal) assumption:

$$\dot{p}_1 = \frac{RT_1}{V_1} (W_{th} - W_{ac}), \quad (3.5)$$

where  $R$  is the ideal gas constant in kJ/kg/K,  $T_1$  is the temperature of the air in the intake manifold in K,  $V_1$  is the intake manifold volume, and  $W_{th}$  is the throttle flow in kg/s. Near idle, the flow through the throttle is choked and one can approximate

$$W_{th} = k_3 u_{th}, \quad (3.6)$$

where  $u_{th}$  is the throttle position in degree and  $k_3$  is a constant. From (3.3) and (3.4) it follows that  $M_{e0} = k_2 p_1 + \frac{k_1}{N} k_0$ . Differentiating this expression and using (3.4)–(3.6) leads to

$$\dot{M}_{e0} = -k_2 \frac{RT_1}{V_1} \frac{N}{k_1} M_{e0} + k_2 \frac{RT_1}{V_1} k_3 u_{th} - \frac{k_0 k_1}{N^2} \dot{N}. \quad (3.7)$$

The term  $\frac{k_0 k_1}{N^2} \dot{N}$  is small and can be omitted so that

$$\dot{M}_{e0} = -k_2 \frac{RT_1}{V_1} \frac{N}{k_1} M_{e0} + k_2 \frac{RT_1}{V_1} k_3 u_{th}. \quad (3.8)$$

From (3.1) and (3.2) it now follows that

$$\dot{N} = \frac{1}{J/(30/\pi)} (M_{e0}(t - t_d) u_\delta - M_L). \quad (3.9)$$

The complete model is now defined by (3.8) and (3.9).

The delay  $t_d$  is between the intake stroke of the engine and torque production, and is about  $360^\circ$  of crankshaft revolution. Consequently,  $t_d$  is inversely proportional to engine speed and is given by

$$t_d(t) = \frac{60}{N(t)}. \quad (3.10)$$

Note that the delay is state-dependent and time-varying since  $N(t)$  varies with time. Nevertheless, in the practical treatment of the problem a constant delay assumption, with the delay value corresponding to that at the idle speed set-point, is often made. Some references (see e.g., [18]) cite a range between  $230^\circ$  and  $360^\circ$  for the delay duration in the crank angle domain rather than a fixed value of  $360^\circ$ , others (see e.g., [14, 44]) augment to  $t_d$  also the throttle delay and computing delay in the engine control module. For instance, in [14] the value of the delay is estimated as  $450^\circ$  for a prototype engine, due to computations scheduling in the engine control module.

It is also interesting that transforming the model to the crank angle domain by defining

$$\theta = \frac{N(t)}{60}t, \quad (3.11)$$

and using the relation  $\frac{d(\cdot)}{d\theta} = \frac{60}{N} \frac{d(\cdot)}{dt}$  renders (3.8)–(3.9) as

$$\frac{dM_{e0}}{d\theta} = \frac{60}{N} \left( -k_2 \frac{RT_1}{V_1} \frac{N}{k_1} M_{e0} + k_2 \frac{RT_1}{V_1} k_3 u_{th} \right), \quad (3.12)$$

$$\frac{dN}{d\theta} = \frac{60}{JN/(30/\pi)} \left( M_{e0}(\theta - 1)u_{\delta} - M_L \right). \quad (3.13)$$

This is a plant model with a constant delay equal to 1.

For a Ford F-150 vehicle, the model parameters in the above model have been estimated as follows [28, 47]:  $k_1 = 3,961$ ,  $k_2 = 102$ ,  $k_3 = 2.02$ ,  $\frac{RT_1}{V_1} = 0.0750$  and  $J/(30/\pi) = 0.027$  while the nominal throttle position, load torque and engine speed are, respectively,  $u_{th,0} = 3.15$ ,  $M_L = 31.15$  and  $N = 800$ . It is of interest to examine the linearized transfer function at idle operating conditions from throttle position in degree to engine speed in rpm. It can be shown that this transfer function has the following form,

$$H(s) = \frac{256.867 \times 2.228}{s^2 + 1.545s + 2.228} e^{-t_d s}.$$

The step response of this transfer function has a lightly damped character (damping ratio about 0.5). The linearized transfer function at idle operating conditions from load torque in Nm to engine speed in rpm has the form

$$H_L(s) = \frac{-37.04s - 57.22}{s^2 + 1.545s + 2.228} e^{-t_d s}.$$

A first-order transfer function with a unit static gain may additionally be augmented to the above model to represent the dynamics of the actual throttle position,  $u_{th}$ , in response to the commanded throttle position,  $u_{th,c}$  set by the idle speed controller. Reasonable parameter estimates for this throttle dynamics model are a static gain of 1, the time constant of about 0.05s and a delay of about 0.05s (if not already accounted in  $t_d$ ).

### 3.2.2 Treating ISC Using Tools for Time-Delay Systems

We now comment on some of the approaches proposed in the literature to treat the delay-related aspects of ISC.

In practical applications, the treatment of the delay using Padé approximations is common. A first-order Padé approximation has the form

$$e^{-st_d} = \frac{e^{-s \frac{t_d}{2}}}{e^{\frac{t_d}{2}}} \approx \frac{-\frac{t_d}{2}s + 1}{\frac{t_d}{2}s + 1}. \quad (3.14)$$

In ISC, such a Padé approximation can be used to represent the delay  $t_d$  between the intake stroke of the engine and torque production. With this approximation, a pole-zero pair is added to the delay-free transfer function model of the plant, thereby permitting the resulting plant model to be treated using conventional control design methods. Note that Padé approximation introduces a right-half plane zero, which renders the system nonminimum phase and tends to guard against excessively high feedback gains that are problematic because of the delay. On the other hand, with the Padé approximation the treatment of the delay is only approximate and the range of the stabilizing gains may be over-predicted [25].

In [38] a controller architecture for two inputs, throttle and spark, is proposed in which the spark input compensates for the delay in the throttle channel. Specifically, a controller is designed first for the throttle to control engine speed based on a nondelayed plant model. Then another controller for the spark timing is proposed to fill in for the difference in engine torque delivery due to the delay, assuming that the spark to torque production delay is neglected. Since the effect of the delay in the throttle to speed channel is compensated by the spark timing, from the perspective of the throttle to engine speed control loop design the delay can be ignored. The effectiveness of this approach has been demonstrated in simulations in [40]; however, the use of spark timing may be restricted in applications where the reduction in spark reserve is sought to improve fuel economy.

In [6] a second-order plus time-delay ISC model is considered, in which the delay  $t_d$  affects throttle to engine speed control channel while the delay in the spark to engine speed control channel is neglected. The model is essentially the same as (3.8)–(3.9), but the physical states in [6] are engine speed and manifold pressure. In addition, time-varying parameter uncertainties (with known bounds) are considered. Since only the engine speed is assumed to be measured, an observer for the full state vector is proposed, in which the output injection term is composed of two gains multiplying, respectively, the error between measured output and estimated output and the error between delayed by  $t_d$  measured output and delayed by  $t_d$  estimated output. A state feedback controller is designed which uses two gains multiplying, respectively, the current and delayed deviations in the state estimate. The design of observer and controller gains is based on the method of Lyapunov-Krasovskii functionals, and it leads to the sufficient conditions, in the form of LMIs, that guarantee the convergence of control and observer errors to zero despite parameter variations. This approach is tested in simulations based on a model of a Ford F-150 vehicle with a 4.6 L V8 engine, in which the delay is assumed to be a constant at 100 ms.

An approach based on the finite spectrum assignment is investigated in [15] for an ISC problem with air-bypass valve and spark timing as inputs. Both delays in air-bypass valve channel and in the spark channel are considered. The model is recast in the form

$$\dot{x}(t) = A_0x(t) + A_1x(t - t_d) + B_0u(t) + B_1u(t - t_s),$$

where  $t_d$  is the delay in the air-bypass valve channel and  $t_s$  is the delay in the spark channel. The state vector  $x(t)$  is four dimensional, with the two physical states

(engine speed and manifold pressure) being similar to the states of the model (3.8)–(3.9), and two additional states appearing because of augmentation of an integrator on the engine speed to ensure zero steady-state error and due to adding a derivative action in the spark channel. The proposed control law has the form

$$u(t) = -Kx(t) - K \int_{t-t_d}^t e^{\tilde{A}(t-t_d-\theta)} A_1 x(\theta) d\theta - K \int_{t-t_s}^t e^{\tilde{A}(t-t_s-\theta)} B_1 u(\eta) d\eta,$$

with the integrals being discretized for implementation using a trapezoidal integration rule. The matrices  $\tilde{A}$  and  $K$  can be determined, under a spectral stabilizability assumption, using a procedure discussed in [15]. The approach was tested in simulations.

Recent work [44] featured an application of Adaptive Posicast controller for time-delay systems developed in [34]. The control law for the throttle has been designed in the form

$$\begin{aligned} u_{th}(t) &= \theta_1^T(t) \omega_1(t) + \theta_2^T(t) \omega_2(t) + \int_{-t_d}^0 \lambda(t, \eta) u_{th}(t + \eta) d\eta + \theta_4(t) r, \\ \frac{\partial \theta}{\partial t} &= -\Gamma(y(t) - y_m(t)) \omega(t - t_d), \\ \theta &= [\theta_1 \ \theta_2 \ \lambda \ \theta_4]^T, \quad \omega = [\omega_1 \ \omega_2 \ u \ r]^T \\ \dot{\omega}_1 &= \Lambda_0 \omega_1 + l u(t - t_d), \\ \dot{\omega}_2 &= \Lambda_0 \omega_2 + l y(t), \end{aligned}$$

where  $y$  is the deviation of the engine speed from the nominal engine speed,  $r$  is the set-point for  $y$ ,  $y_m$  is the output of the reference model,  $\Lambda_0$  and  $l$  are obtained in the design [44] and  $\Gamma$  is an adaptation gain. The Adaptive Posicast controller guarantees that the closed-loop system is stable and that the output of the plant follows the output of the reference model. The integral term has been discretized according to

$$\int_{-t_d}^0 \lambda(t, \eta) u(t + \eta) d\eta \approx \lambda_1 u_{th}(t - dt) + \dots + \lambda_k u_{th}(t - k dt),$$

where  $dt$  is the sampling interval and  $k dt = t_d$ . The controller was implemented experimentally in a Ford F-150 vehicle and its capability to outperform the existing production controller in set-point following and disturbance rejection has been demonstrated. This improved performance comes at a price of increased computational complexity. Potential opportunities to reduce calibration time and effort through the application of this adaptive control approach have been also highlighted in [44].

As yet another application of the recent results in the literature on time-delay systems, [49] considered an observer design problem for the states of a system affected by random and unknown delays. The delay is modeled by a Markov chain with a finite number of states, and the observer structure has been proposed based on a probabilistically averaged copy of the state space model. The observer was

successfully applied as a fault detector for an engine speed control system in which the control input (requested engine torque) is computed in a module connected to engine control module via a controller area network (CAN).

### 3.3 Air-to-Fuel Ratio Control

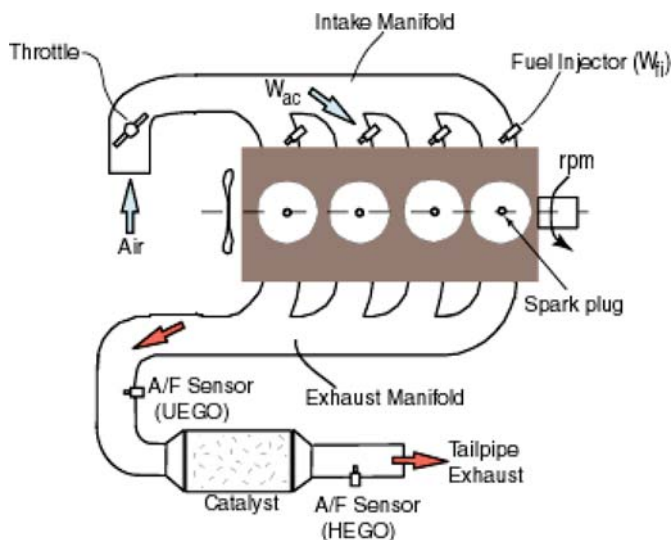
Air-to-fuel ratio control is imperative for conventional gasoline vehicles (see Fig. 3.2) in order to achieve emission reductions to mandated levels. The original introduction of the electronic powertrain control in the late 1970s has, in fact, been largely driven by the air-to-fuel ratio control requirement stemming from the introduction of tighter emission standards for the passenger vehicles. The conversion efficiency of a three-way catalyst (TWC) as a function of the tailpipe air-to-fuel ratio<sup>1</sup> is rapidly degraded when the tailpipe air-to-fuel ratio deviates from the stoichiometry [17]. Because of oxygen storage in the catalyst, the tailpipe air-to-fuel ratio (after the catalyst) can remain at stoichiometry even if the feed-gas air-to-fuel ratio (before the catalyst) deviates from stoichiometry in transients. In steady-state conditions the two air-to-fuel ratios equilibrate. Thus the catalyst provides a capability to withstand some transient fluctuations in the feed-gas air-to-fuel ratio without leading to emission performance degradation.

The feed-gas air-to-fuel ratio can be disturbed by engine airflow transients, when the driver tips-in or tips-out, or by periodic canister vapor purges, which result in extra air and fuel vapor flow from the carbon canister that stores fuel vapors originating in the fuel tank. Tight control of the feed-gas air-to-fuel ratio to its set-point (this set-point may depend on an estimate of the stored oxygen in the catalyst or on an indication/measurement of the tailpipe air-to-fuel ratio) is imperative to reduce emissions or to achieve reductions in the cost and size of the catalyst. The feed-gas air-to-fuel ratio is measured by a wide-range air-to-fuel ratio sensor, which is also often referred to as the universal exhaust gas oxygen (UEGO) sensor. The tailpipe air-to-fuel ratio may be measured either by a wide-range sensor or, more typically, by a switching sensor, called heated exhaust gas oxygen (HEGO) sensor, which provides an indication if the tailpipe air-to-fuel ratio is richer or leaner than stoichiometry. The HEGO sensor also has some resolution in a narrow range around the stoichiometry, i.e., its voltage reading is sensitive to the actual air-to-fuel ratio value but in a narrow range and with a steep slope. The HEGO sensor may be used to measure the tailpipe air-to-fuel ratio because it is normally cheaper, faster, and is less subject to drift as compared with the UEGO sensor. Moreover, if the engine is properly controlled, the tailpipe air-to-fuel ratio tends to vary in a much narrower range than the feed-gas air-to-fuel ratio.

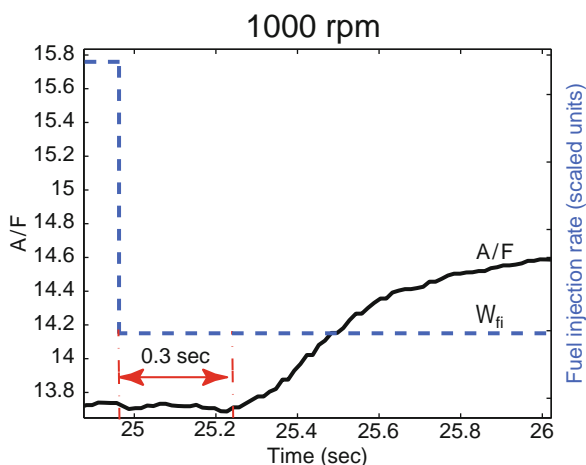
The delay between fuel injection and UEGO sensor measurement, see Fig. 3.3 for an illustration, can be a limiting factor seriously degrading the achievable

---

<sup>1</sup> The air-to-fuel ratio is a quantity reflecting the exhaust gas composition in terms of equivalent amounts of air and fuel which need to be burnt.



**Fig. 3.2** Schematics of a gasoline engine



**Fig. 3.3** Delay from fuel injection step to feed-gas air-to-fuel ratio change at 1,000 rpm for an experimental engine

performance of the air-to-fuel ratio feedback loop. Thus more effective techniques for controlling time delay systems can have a substantial impact in reducing conventional gasoline vehicle tailpipe emissions. While our subsequent treatment is focused on conventional gasoline engines, we note that the delay also plays an important role in the air-to-fuel ratio control for advanced lean-burn engines and for diesel engines. We refer the reader to [1] and [48] for the design of the air-to-fuel ratio controller while accounting for the delay.

In the remainder of this section we first discuss a typical air-to-fuel ratio control loop for a gasoline engine based on a wide range air-to-fuel ratio sensor, with special

attention paid to the delay modeling. We will then illustrate an application of certain tools for characterizing the range of stable gains in this control loop when the airflow is constant and we will discuss links with recently developed theoretical tools for control of time-delay systems which permit to treat the case of time-varying air flow. The case of air-to-fuel ratio control loop based on a switching air-to-fuel ratio (HEGO) sensor rather than the wide range air-to-fuel ratio (UEGO) sensor will be briefly discussed at the end.

### 3.3.1 Feed-gas Air-to-Fuel Ratio Control Block Diagram

A typical arrangement of a closed-loop air-to-fuel ratio system, based on proportional-plus-integral control and a wide range air-to-fuel ratio sensor, is shown in Fig. 3.4. As in our treatment of ISC, cyclic fluctuations of engine operating variables are replaced by cycle averages using a mean-value approximation so that the resulting plant and control system dynamics can be treated in continuous-time. The injected fuel flow rate (control input) is denoted by  $W_{fi}$ . A fraction of the injected fuel enters the cylinder during the intake event in vapor form, while the remainder in liquid form replenishes a fuel puddle formed in the intake ports of the engine. The liquid fuel in the puddle evaporates and enters the cylinder in vapor form. Thus, in steady-state conditions the amount of fuel entering the cylinder is equal to the amount of injected fuel, but in transient conditions these two quantities can be different. The transient fuel dynamics are shown by the block “TF” on the diagram in Fig. 3.4. Additional fuel and air can enter as a disturbance,  $d$ , due to canister vapor purge. The in-cylinder fuel-to-air ratio is  $\mu = \frac{W_{fc}}{W_{ac}}$ , where  $W_{ac}$  denotes the air flow into the engine cylinders. A first-order transfer function plus time delay represents the dynamics between actual fuel-to-air ratio and measured fuel-to-air ratio. The time constant,  $\tau_e$ , accounts for the fact that individual cylinders do not fire simultaneously within the cycle [27], for the exhaust mixing and for the UEGO sensor time constant. The delay,  $t_d$ , mainly accounts for the intake to exhaust stroke

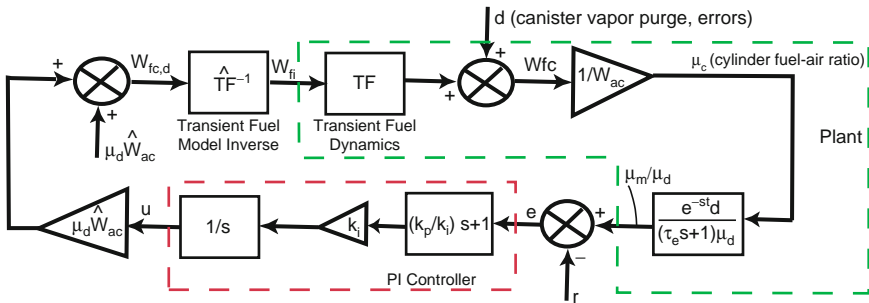


Fig. 3.4 Block diagram of gasoline engine air-to-fuel ratio control system

engine cycle delay and for the transport delay to exhaust sensor location; it will be considered in Sect. 3.3.2 in more detail. As will be detailed below, the delay  $t_d$  varies with engine operating conditions and it is dependent on engine speed and engine air flow.

The closed-loop regulation to a desired fuel-to-air ratio,  $\mu_d$ , is typically performed by a proportional-plus integral (PI) controller. The value of  $\mu_d$  is around  $1/14.64$  in applications to gasoline stoichiometric engines and may be adjusted by another (outer) control loop based on either voltage reading from a tailpipe air-to-fuel ratio sensor (HEGO) or based on estimated oxygen storage in the catalyst. Other control designs, such as an  $H_\infty$  controller (see e.g., [1, 36]) or Adaptive Posicast Control (see [45, 46]) can, of course, be also used. The PI controller gains  $k_p$  and  $k_i$  can be coordinated to cancel the plant pole at  $-1/\tau_c$  [27], and for that reason, the term  $\frac{k_p}{k_i}s + 1$  is factored out. The estimate of the airflow multiplied by desired fuel-to-air ratio,  $\mu_d \hat{W}_{ac}$  scales the PI controller output,  $u$ , i.e.,  $u$  provides a multiplicative correction to the feedforward fuel quantity,  $\mu_d \hat{W}_{ac}$ . The desired in-cylinder fueling rate is thus  $W_{fc,d} = \mu_d \hat{W}_{ac}(1 + u)$ . Since the engine airflow varies over a broad range, a multiplicative correction, which scales automatically with the airflow, results in less transient air-to-fuel ratio excursions in fast transients. Furthermore, it renders the static (dc) gain of the plant constant across operating range. The transient fuel model inverse block, “TF<sup>-1</sup>”, backtracks the required injected fueling rate,  $W_{fi}$ , to realize  $W_{fc,d}$ .

### 3.3.2 Modeling

While most of the blocks in Fig. 3.4 are self-explanatory and can thus be easily modeled, the model of the delay,  $t_d$ , and of the transient fuel dynamics deserve further discussion.

#### 3.3.2.1 Delay

The delay between fuel injection and fuel-to-air measurement is comprised of the delay between fuel injection and intake event,  $t_{fi}$ , delay between intake and exhaust event,  $t_{ie}$ , transport delay to the A/F sensor location,  $t_{tr}$ , sensor delay,  $t_s$  and computational delay in the engine control module,  $t_c$ . Assuming two events between fuel injection on the closed valve and engine intake event and a four cylinder engine, the following approximations can be made

$$t_{fi} = 2 \frac{30}{N}, \quad t_{ie} = 3 \frac{30}{N},$$

where  $N$  denotes the engine speed in rpm. If the length of travel of the exhaust gas to the A/F sensor location is  $L$ , then,

$$t_{tr} = \frac{L}{v},$$

where  $v$  is the gas velocity. From the ideal gas law and using the definition of the mass flow rate of the exhaust,  $W_{\text{ex}}$ , we obtain

$$W_{\text{ex}} = \frac{p_2}{RT_2}Av,$$

where  $p_2$  is the exhaust pressure,  $R$  is the gas constant,  $T_2$  is the exhaust gas temperature, and  $A$  is the flow area. Thus,

$$t_{\text{tr}} = \frac{Lp_2A}{RT_2W_{\text{ex}}}.$$

In a naturally aspirated stoichiometric port-fuel injected engine,  $p_2$  is approximately constant while the heat flux,  $T_2W_{\text{ex}}$ , is essentially proportional to the mass fuel rate, which in turn is proportional to the mass flow rate of air,  $W_{\text{ac}}$ . Thus the total delay,  $t_c + t_{\text{fi}} + t_{\text{ie}} + t_{\text{tr}} + t_s$ , is mainly a function of engine speed,  $N$ , and the mass flow rate of air,  $W_{\text{ac}}$ .

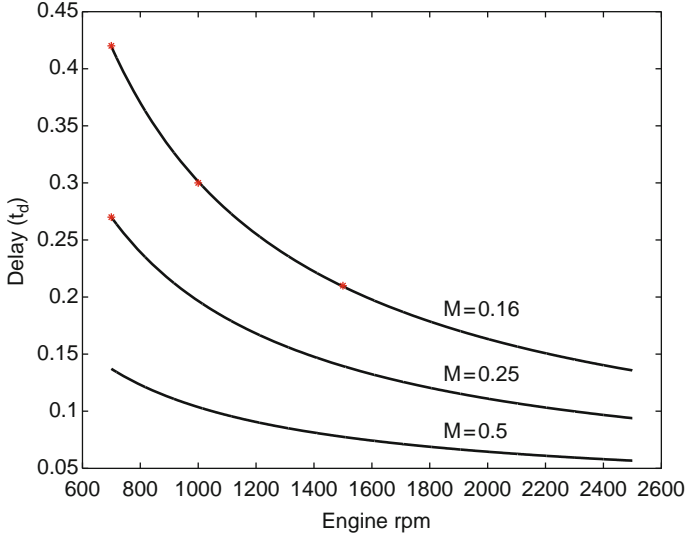
In contrast to ISC, the delay in the air-to-fuel ratio control problem cannot be assumed to be constant since engine speed and air flow, on which it depends, are varying widely across engine operating range. On the other hand, the delay is not dependent on the states of the air-to-fuel ratio control problem itself.

Recent measurements taken from a Ford vehicle with a 5.4 L V8 engine [32] have been fitted into an aggregate model derived from the above considerations:

$$t_d = \alpha_0 + \frac{\alpha_1}{N} + \frac{\alpha_2}{MN}, \quad (3.15)$$

where  $N$  is the engine speed,  $M$  is the load, and  $\alpha_0 = 0.0254$ ,  $\alpha_1 = -14.7798$ ,  $\alpha_2 = 46.4974$ . The load  $M$  is defined as the ratio of  $W_{\text{ac}}/N$  to maximum possible  $W_{\text{ac}}/N$  at a given engine speed, if the engine is operated at sea level and the flow of air into the cylinder is at normal ambient (25°C) temperature. For the engine on which the delay measurements were conducted, the mass flow rate of air,  $W_{\text{ac}}$  (in kg/s) as a function of engine speed and load is given by  $W_{\text{ac}} = 5.53 \times 10^{-5}MN$ . Figure 3.5 illustrates the delay dependence on engine speed and load.

With  $\tau_c$  about 0.2 (typical value), the product of the pole at  $a = \frac{1}{\tau_c}$  and the delay,  $t_d$ , is larger than 0.5 if the delay is larger than 0.1. In process control literature, first-order plus time-delay systems with  $at_d \geq 0.5$  are considered to be difficult to control using only feedback and feedforward compensation is typically required [26]. Perhaps as a manifestation of this difficulty, in air-to-fuel ratio control applications accurate air charge estimation, accurate transient fuel compensation, and estimation of purge flow disturbance are employed to realize effective control. The development of these feedforward estimates entails calibration time and effort, which may be reduced if more effective methods are applied to the design of the feedback compensation. Note also that in other cases, e.g., in lean-burn engine applications considered in [48], the control system uses tailpipe air-to-fuel ratio measurement directly, with the need to treat a significantly larger time delay.



**Fig. 3.5** The delay as a function of engine speed for different loads. Asterisks indicate experimental engine measurements

### 3.3.2.2 Transient Fuel Modeling

Several models of varying complexity have been developed in the literature to model transient fuel behavior. The original mean-value model [3] has the following form:

$$\begin{aligned} \frac{dm_{fp}}{dt} &= -\frac{m_{fp}}{\tau} + XW_{fi}, \\ W_{fc} &= (1 - X)W_{fi} + \frac{m_{fp}}{\tau}, \\ \lambda &= \frac{W_{ac}}{W_{fc}}, \end{aligned} \quad (3.16)$$

where  $m_{fp}$  denotes the mass of the fuel in the liquid fuel puddle formed in the ports (total for all cylinders),  $\tau > 0$  is the fuel evaporation rate from the liquid puddle, and  $X$  is a fraction of the injected fuel that replenishes the puddle ( $0 < X < 1$ ). The parameter  $X$  is primarily a function of the engine coolant temperature (which changes slowly and is a measured quantity) but  $\tau$  exhibits a more complicated dependence on the engine operating variables, for instance, it may depend on the intake manifold pressure, air flow and engine speed. The inverse of this transient fuel model has the following form,

$$W_{fi} = \frac{W_{fc,d} - \frac{\hat{m}_{fp}}{\tau}}{1 - X}, \quad (3.17)$$

where  $\hat{m}_{fp}$  is an estimate of liquid fuel in the puddle. For more elaborate transient fuel models, see, for instance, [24] and references therein.

### 3.3.3 Constructing Stability Charts and Gain Scheduling

The analysis of stability and convergence rates for the air-to-fuel ratio control system in Fig. 3.4 in the case when the engine speed,  $N$ , and air flow into the engine,  $W_{ac}$  are constant is of interest as it characterizes the air-to-fuel ratio behavior after a transition (such as a tip-in or a tip-out) has occurred to a new operating condition. In this case,  $t_d$  and  $\tau_e$  are constant. Assuming accurate transient fuel compensation and in absence of the canister vapor purge disturbances, the error,  $e$ , can be shown to satisfy the following equations,

$$\begin{aligned} \dot{z} &= e, \\ \dot{e} &= -\frac{1}{\tau_e}e(t) - \frac{1}{\tau_e} \left( k_p e(t - t_d) + k_i z(t - t_d) \right). \end{aligned} \quad (3.18)$$

To determine the range of the gains,  $k_p$  and  $k_i$ , for which the closed-loop stability and a specified convergence rate are guaranteed, the numerical method of [9] may be used. This method is applicable to general linear time-periodic delay-differential equations with multiple delays commensurate to the period. The system (3.18) is a linear time-invariant system with a single delay and can thus be handled by the method in [9] with ease.

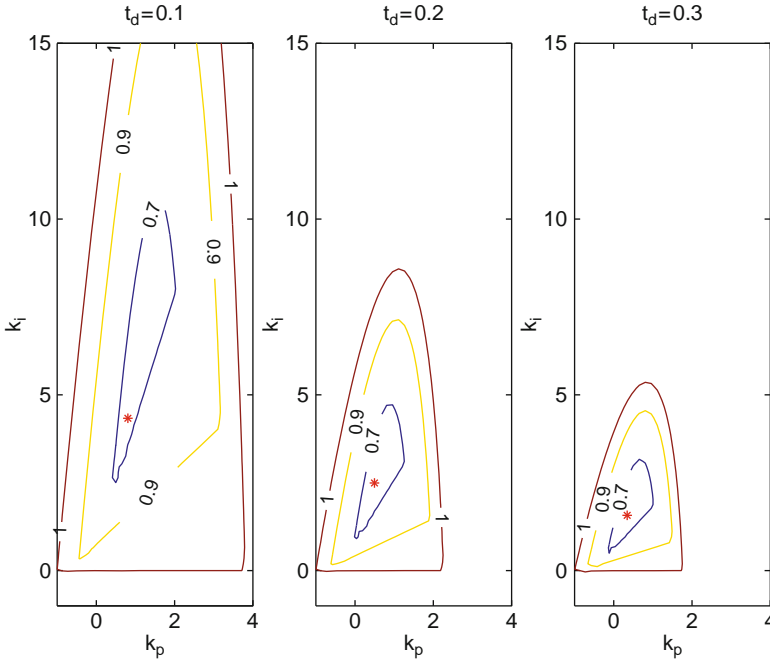
To apply the approach of [9] it is convenient to transform (3.18) into a form

$$\begin{aligned} \frac{d}{d\theta}x &= A(\varepsilon)x(\theta) + B(\varepsilon)x(\theta - 1), \\ x &= [z, e]^T, \quad \varepsilon = [t_d, \tau_e, k_p, k_i]^T, \quad \theta = \frac{t}{t_d}, \end{aligned} \quad (3.19)$$

where  $\theta$  is the scaled time and  $\varepsilon$  is a parameter vector. Consistently with the method of steps [30], (3.19) may be viewed as defining a linear operator on  $C[-1, 0]$  that maps past solution segments to future solution segments. By approximating functions on  $[-1, 0]$  using interpolation through Chebyshev extreme points, a finite-dimensional linear operator,  $U(\varepsilon)$ , is obtained, which maps the coefficients of the associated Chebyshev interpolating polynomials. The Chebyshev basis is selected as it combines spectral accuracy with simplicity of computations. To compute an approximation to the stability region in a parameter space, one needs to choose a grid for  $\varepsilon$  and compute the eigenvalues and spectral radius,  $\rho(\varepsilon)$ , of  $U(\varepsilon)$ , for each grid point. The stability region approximately corresponds to the values of  $\varepsilon$  for which  $\rho(\varepsilon) < 1$  and its boundary corresponds to  $\rho(\varepsilon) = 1$ .

The guaranteed convergence rate regions can be defined using  $\rho(\varepsilon)$  as an indicator of the convergence rate, i.e.,  $\rho(\varepsilon) \leq \rho_0$  for a fixed  $\rho_0 < 1$ . The smaller  $\rho_0$  is, the faster the convergence rate. Since a time transformation is employed to arrive at (3.19), a better indicator of the convergence rate in time can be defined as follows:

$$\bar{\rho}(\varepsilon) = -\frac{t_d}{\log \rho(\varepsilon)}. \quad (3.20)$$



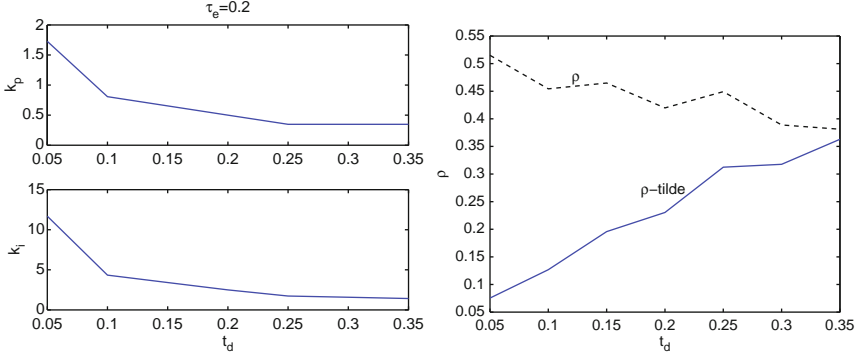
**Fig. 3.6** The stability and guaranteed convergence rate regions (given in terms of  $\rho$ ) in the  $k_p - k_i$  parameter space for different values of  $t_d$  and  $\tau_e = 0.2$

The  $\tilde{\rho}(\varepsilon)$  may be viewed as an equivalent time constant of the closed-loop system. The Matlab<sup>2</sup> software implementing the computations is located at <http://www.cs.uaf.edu/~bueler/ddepage.htm>.

Figure 3.6 illustrates the stability and guaranteed convergence rate regions in terms of  $k_p$  and  $k_i$  for  $t_d = 0.1, 0.2, 0.3$  and for  $\tau_e = 0.2$ . We used 25 Chebyshev extreme points in constructing  $U(\varepsilon)$ . Notice that the regions shrink considerably with the increase in the delay. Figure 3.7–left presents the values of  $k_p$  and  $k_i$ , which minimize  $\rho$  as a function of  $t_d$  and suggests that the values of  $k_p$  and  $k_i$  can be gain-scheduled as a function of  $t_d$  to preserve the stability and maximize the convergence rate. Figure 3.7–right compares the minimum values of  $\rho$  and  $\tilde{\rho}$  for optimum gains as a function of the delay  $t_d$ . Note that the achievable values of  $\tilde{\rho}$  are increasing with  $t_d$ , which is an indication of the larger delay limiting more the performance of the closed-loop system.

In [4] a similar air-to-fuel ratio control system is analyzed and it is shown that the numerical boundaries of the stability and guaranteed convergence rate regions are very close to the analytically predicted boundaries.

<sup>2</sup> Matlab is a registered trademark of the Mathworks, Inc., Natick, MA.



**Fig. 3.7** *Left:* The values of  $k_p$  and  $k_i$ , which minimize  $\rho$  as functions of  $t_d$ . *Right:* The values of  $\rho$  and  $\tilde{\rho}$  as functions of  $t_d$

### 3.3.4 Using Robust and Stochastic Stabilization Tools for Time-Delay Systems

Our treatment in Sect. 3.3.3 has been based on the constant air flow assumption. Robust and stochastic stability analysis tools for time-delay systems can be used to treat a more general case, and we now provide a brief illustration of this for an air-to-fuel ratio controller of feedback linearization type originally proposed in [4].

With  $\mu = \frac{W_{fc}}{W_{ac}}$  denoting the in-cylinder fuel-to-air ratio, it can be shown from (3.16) and the assumption that  $X$  does not vary with time (since  $X$  is mainly a function of slowly varying engine coolant temperature) that

$$\dot{\mu} = \frac{1-X}{W_{ac}} \dot{W}_{fi} + \frac{W_{fi}}{\tau W_{ac}} - \mu \delta + \frac{(1-X)W_{fi}}{W_{ac}} \frac{\dot{\tau}}{\tau}, \quad (3.21)$$

where

$$\delta = \frac{1}{\tau_0} + \tilde{\delta}, \quad \tilde{\delta} = \left( \frac{1}{\tau} - \frac{1}{\tau_0} \right) + \frac{\dot{\tau}}{\tau} + \frac{\dot{W}_{ac}}{W_{ac}},$$

and  $\tau_0$  is a nominal value of  $\tau(t)$ . With  $\mu_m$  denoting the measured fuel-to-air ratio,  $\mu_d$  being the desired fuel-to-air ratio and defining  $\xi = \mu - \mu_d$ ,  $\xi_m = \mu_m - \mu_d$ , we obtain that

$$\dot{\xi}_m + a \xi_m = a \xi(t - t_d), \quad a = \frac{1}{\tau_e}. \quad (3.22)$$

While the air flow,  $W_{ac}$ , and fuel evaporation rate parameter  $\tau$  are considered as time-varying, two simplifying assumptions, of constant  $t_d$  and constant  $a$ , are now made. These two assumptions may be justified as approximations in the analysis of the engine operating only in a part of the full operating range (e.g., during constant speed cruise with limited load changes); further, a conservative approach of augmenting a time-varying delay in software in the engine control module can effectively render  $t_d$  constant. The treatment of a more general model (with

time-dependent  $t_d$  and  $a$ ) is possible along similar lines but the notations and the required conditions are considerably more cumbersome [12] and are not considered here.

With these assumptions, it follows that

$$\ddot{\xi}_m(t + t_d) + a\dot{\xi}_m(t + t_d) = a \left[ \frac{1-X}{W_{ac}} \dot{W}_{fi} + \frac{W_{fi}}{\tau W_{ac}} - \mu_d \delta - \xi \delta + \frac{(1-X)W_{fi}}{W_{ac}} \frac{\dot{\tau}}{\tau} \right].$$

We note that this equation can be used as a basis for system identification of  $a$ ,  $X$ ,  $\tau$ , and  $t_d$ , see [16]. Since, per process control guidelines, we established that the air–fuel ratio plant is difficult to control using feedback-only, accurate system identification is very important and there is much room for the development and application of effective system identification procedures for time-delay systems. In this regard, and given that the delay in the air-to-fuel ratio control problem is an output delay and not the state delay, the convolution methods in [5] appear promising for fast time-delay identification. Further treatment of identification issues for time delay systems falls beyond the scope of this chapter.

With a feedback-linearizing type controller for the fuel-injection rate,  $W_{fi}$ , defined according to

$$\frac{1-X}{W_{ac}} \dot{W}_{fi} + \frac{W_{fi}}{\tau W_{ac}} - \mu_d \delta + \frac{(1-X)W_{fi}}{W_{ac}} \frac{\dot{\tau}}{\tau} = \frac{k}{a} \xi_m, \quad (3.23)$$

the closed-loop dynamics are

$$\ddot{\xi}_m(t) + \left( a + \frac{1}{\tau_0} + \tilde{\delta}(t - t_d) \right) \dot{\xi}_m(t) + \left( \frac{a}{\tau_0} + a\tilde{\delta}(t - t_d) \right) \xi_m - \frac{k}{a} \xi_m(t - t_d) = 0. \quad (3.24)$$

Figure 3.8 shows the experimental response of the controller (3.23) operating the engine at 1,500 rpm and 20°C engine coolant temperature, where because of the cold temperature the air-to-fuel ratio control is particularly challenging. The air flow into the engine is changing consistently with the torque ramp up and ramp down and the performance is evaluated with an estimate of  $\dot{W}_{ac}$  included into (3.23) but with  $\tau$  set to a constant value. As the responses show, including an estimate of  $\dot{W}_{ac}$  and the feedback due to  $k \neq 0$  enhance the controller disturbance rejection properties.

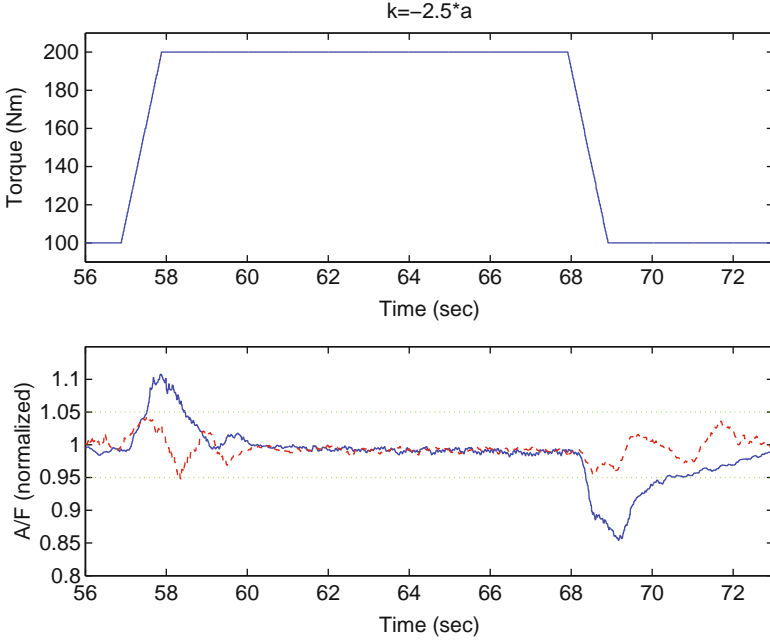
Note that the term  $\tilde{\delta}(t - t_d)$  may be viewed as a time-varying uncertainty in the coefficients of (3.24), so that the robust or stochastic stability of (3.24) may be analyzed by the LMI type methods developed in the literature, see e.g. [7] and [30]. With  $x_1 = \xi_m$ ,  $x_2 = \dot{\xi}_m$ , and

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{a}{\tau_0} & -(a + \frac{1}{\tau_0}) \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ k & 0 \end{bmatrix},$$

$$F = \begin{bmatrix} 0 & 0 \\ -a \cdot \theta & -\theta \end{bmatrix},$$

and

$$\tilde{\delta}(t - t_d) = \theta \Delta(t), |\Delta(t)| \leq 1,$$



**Fig. 3.8** Experimental results: Feedback linearization controller performance with the air flow disturbance for  $k = 0$ ,  $\hat{W}_{ac} = 0$  (solid, blue) and for  $k/a = -2.5$  and  $\hat{W}_{ac} \approx \dot{W}_{ac}$  (dashed, red), where  $\hat{W}_{ac}$  is the estimate of the air flow time rate of change. The commanded normalized air-to-fuel ratio is 1 and the 5% deviation band is shown by dashed green lines

the closed-loop model can be recast as

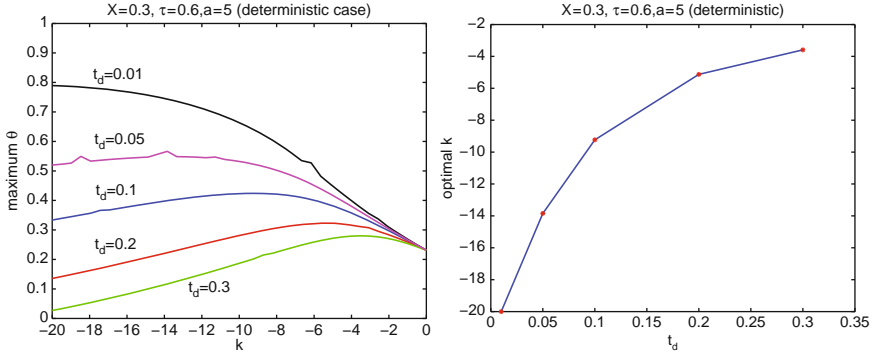
$$\dot{x} = (A + \Delta(t)F)x(t) + Bx(t - t_d), \quad (3.25)$$

where  $A$ ,  $B$ , and  $F$  are system matrices and  $\Delta(t)$  is a scalar time-varying uncertainty satisfying  $|\Delta(t)| \leq 1$ .

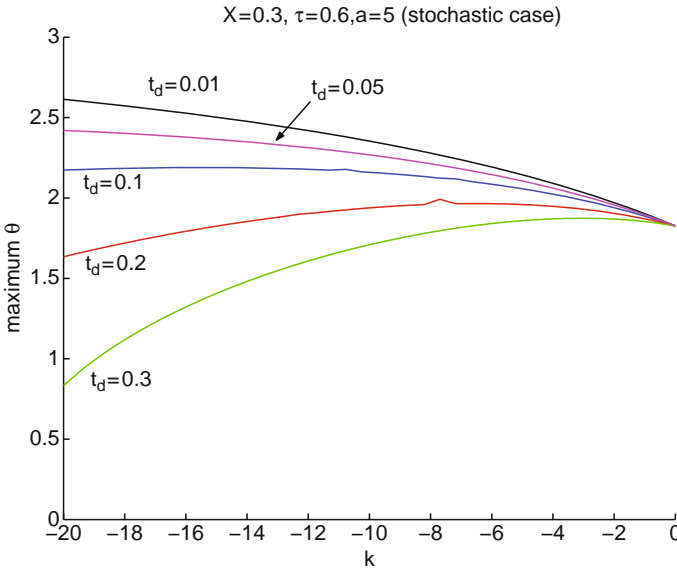
The LMI-based conditions, reviewed in the Appendix, can be used to conservatively estimate the maximum size of  $\theta$ , which preserves closed-loop stability for different  $k$  and  $t_d$ . The maximum tolerable  $\theta$  and the gain,  $k$ , for which tolerable  $\theta$  is maximized are shown in Fig. 3.9.

We may instead choose to view  $\tilde{\delta}$  as a stochastic process and proceed to characterize the mean-square stochastic stability of (3.24). The conditions are given by (3.52) in the Appendix where we assume<sup>3</sup> that  $\tilde{\delta} \sim \theta w$ , and  $w$  is a standard Wiener process. We estimate numerically the maximum tolerable value of  $\theta$ , as a function of  $k$  and  $t_d$ . Figure 3.10 signifies that the use of feedback helps to accommodate larger  $\theta$  although larger delays decrease significantly the maximum tolerable value

<sup>3</sup> This means that the change in  $\tilde{\delta}$  from one sample to the next divided by square root of the sampling period is assumed to behave approximately as a zero mean gaussian random variable with variance  $\theta$ .



**Fig. 3.9** *Left*: Maximum tolerable deterministic uncertainty size,  $\theta$ , as a function of the gain  $k$  for different values of  $t_d$ . *Right*: The gain  $k$  that accommodates the maximum deterministic uncertainty size as a function of the delay,  $t_d$



**Fig. 3.10** Solid lines show maximum admissible value of  $\theta$  in the stochastic case. A slight “blip” in  $t_d = 0.2$  trajectory is due to numerical problems

of  $\theta$ . For a standard driving cycle used in vehicle emissions and fuel economy certification, further analysis has revealed that the rate of change of  $\tilde{\delta}$  may be larger than conservative sufficient conditions in Fig. 3.9 permit, but it appears to satisfy conditions of Fig. 3.10.

Continuing now with the analysis of the closed-loop behavior, let  $\hat{X}$ ,  $\hat{\tau}$ ,  $\hat{W}_{ac}$  denote, respectively, the estimates of  $X$ ,  $\tau$ ,  $W_{ac}$  used in controller implementation. Then in steady-state conditions,

$$\mu = \mu_d \left( \frac{1 - \frac{\hat{\tau}k}{a}}{\frac{W_{ac}}{\hat{W}_{ac}} - \frac{\hat{\tau}k}{a}} \right).$$

Thus in terms of steady-state operation, air flow estimation errors are more troublesome for the controller while  $\hat{X}$ ,  $\hat{\tau}$  differences from  $X$ ,  $\tau$  do not lead to a steady-state offset of  $\mu$  from  $\mu_d$  if  $\hat{W}_{ac} = W_{ac}$ . Large magnitude of  $k$  helps to reduce the steady-state offset due to air flow estimation errors but the gain increase is restricted because of the delay and sensor noise.

### 3.3.5 Air-to-Fuel Ratio Control Using a Switching HEGO Sensor

The air-to-fuel ratio regulation can be performed using a switching (HEGO) air-to-fuel ratio sensor in place of a wide range air-to-fuel ratio sensor before the catalyst, see Fig. 3.2. The advantages of a HEGO sensor are lower cost, faster response, and higher tolerance to drift, while the main disadvantage is that for air-to-fuel ratio outside of a narrow range; it only provides an indication of whether the air-to-fuel ratio is rich or lean of stoichiometry, but not by how much.

A model relevant to the air-to-fuel ratio regulation with the HEGO sensor has the following form:

$$\begin{aligned} \dot{s} + as &= a \left( \frac{W_{fc}}{W_{ac}} - \mu_d \right) + d, \quad a = \frac{1}{\tau_e}, \\ y_m &= \text{sign}(s(t - t_d)), \end{aligned} \quad (3.26)$$

where  $y_m$  is the measurement by HEGO sensor,  $\mu_d$  is the fuel-to-air ratio set-point, and  $d$  denotes the disturbance. We treat  $W_{fc} = W_{fc,d}$  (commanded cylinder fueling rate) as a control input, recognizing that the fuel injection rate  $W_{fi}$  is backtracked from commanded  $W_{fc}$  using transient fuel inverse model while the errors in realizing  $W_{fc}$  are lumped into  $d$ . In the subsequent analysis, we will focus on the air-to-fuel ratio regulation under the assumption of air flow being a constant parameter.

The controller has the following form,

$$\begin{aligned} W_{fc} &= W_{ac} \left( \hat{\mu}_d + \hat{s} + \frac{v}{a} \right), \quad \dot{\hat{s}} + a\hat{s} = a \left( \frac{W_{fc}}{W_{ac}} - \hat{\mu}_d \right), \\ v &= -\gamma y_m = -\gamma \text{sign}(s(t - t_d)), \end{aligned} \quad (3.27)$$

where  $\gamma > 0$  is a control gain and in the analysis we assume that<sup>4</sup>  $\hat{\mu}_d = \mu_d$ . It is straightforward to see that (3.27) is a proportional-plus-integral controller for  $W_{fc}$  applied to the measurement  $y_m$  and augmented with a feedforward control. The proportional gain of the controller is equal to  $W_{ac}\gamma/a$  and the integral gain is equal

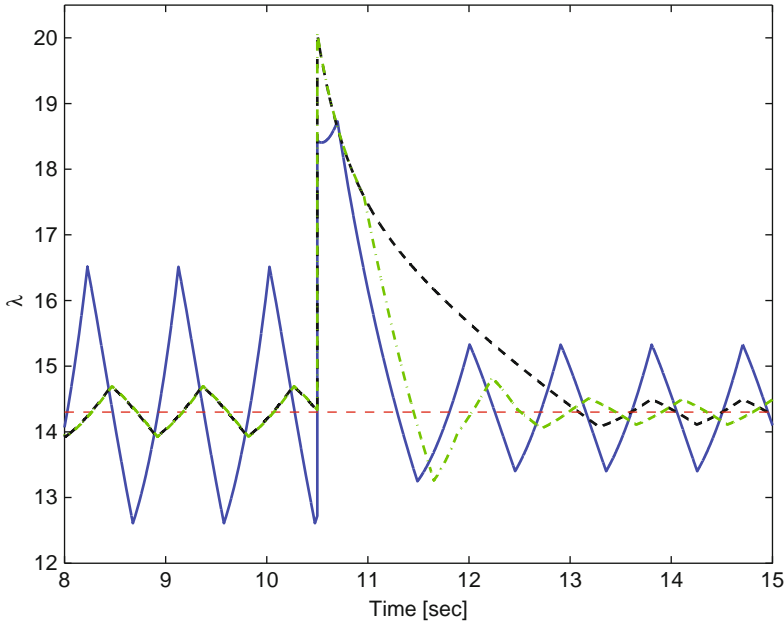
<sup>4</sup> The HEGO sensor switches around the true stoichiometry  $\mu_d$ , while  $\hat{\mu}_d$  is used by the controller in (3.27). Depending on the composition, values of stoichiometric air-to-fuel ratio may vary for gasoline or gasoline-ethanol blend.

to  $W_{ac}\gamma$  so that the gains are coordinated to achieve pole-zero cancellation of the time constant  $\frac{1}{a} = \tau_e$  in the plant model with the term  $(\frac{k_p}{k_i}s + 1)$  of the PI controller, cf. Fig. 3.4.

We illuminate the behavior of the closed-loop system via simulations where large parameter uncertainty has been introduced. The engine parameters are  $X = 0.3$ ,  $\tau = 0.2$ ,  $t_d = 0.2$ ,  $\mu_d = 1/14.3$ , and the air flow into the engine  $W_{ac}$  increases from 40 kg/hr to 80 kg/h at the time instant  $t = 10.5$  s. The controller uses estimates of the parameters,  $\hat{X} = 0.2$ ,  $\hat{\tau} = 0.3$ ,  $\hat{t}_d = 0.23$ ,  $\hat{\mu}_d = 1/14.64$ , and the estimate of the air flow into the engine,  $\hat{W}_{a,c}$ , which increases from 40 kg/h to 60 kg/h. Figure 3.11 shows the closed-loop system behavior for two values of the feedback gain. Even though the parameters are not estimated correctly, the controller is able to cope with parameter uncertainties. The smaller value of the gain results in a smaller amplitude of the limit cycle but is much slower in compensating the transient at  $t = 10.5$  s. On the other hand, the larger value of the gain quickly compensates the transient at  $t = 10.5$  s, but results in a large amplitude limit cycle in steady-state conditions. Thus, none of the two choices of the gain is ideal. Better performance may be attained if the gain is made time-varying via adaptation as discussed next.

From (3.26)-(3.27), we obtain

$$\begin{aligned} \dot{s}(t) &= -\gamma \text{sign}(s(t - t_d)) + w(t), \\ w(t) &= d(t) + a(\hat{s} - s). \end{aligned} \quad (3.28)$$



**Fig. 3.11** Time history of the air-to-fuel ratio for the controller with the high fixed gain (*solid*), for the controller with the low fixed gain (*dashed*) and for the controller with the adaptive gain (*dash-dot*)

Note that if  $e = \hat{s} - s$ , then  $\dot{e} = -ae - d$ . Letting  $z = \dot{e}$ , we have  $\dot{z} = -az - \dot{d}$  and  $d$  slow-varying implies  $w = -z = d + ae$  is ultimately bounded in an interval  $[-1, 1] \cdot \frac{\sup_t |d(t)|}{a}$ . If  $d$  is fast-varying but bounded, then  $w(t)$  is ultimately bounded in an interval  $\sup_t |d(t)|(1 + \frac{1}{a})[-1, 1]$ . In either case, we can treat  $w(t)$  as a bounded signal, i.e.,

$$\sup_t |w(t)| \leq \varepsilon, \quad (3.29)$$

where  $\varepsilon > 0$  is a known bound.

It is interesting to note that (3.28) is a relay-type time-delay system of the kind recently studied in the time-delay literature, see e.g., [10]. Unlike the case treated in [10], here the dynamics with  $\gamma = 0$  are marginally stable and not unstable and there is a bounded disturbance. Nevertheless similar ideas to [10] of basing the gain reduction on predicted zero crossings [11] can be exploited here.

To reduce the amplitude of the steady-state limit cycle in the air-to-fuel ratio trajectory the basic idea is to make the gain  $\gamma$  time-varying,  $\gamma = \gamma(t)$ , and reduce its values at time instants when  $s(t)$  is predicted to cross zero. Specifically, if at time  $t_{k-1}$  a zero crossing by  $s_m(t) = s(t - t_d)$  is detected, we can estimate a predicted time of next zero crossing by  $s(t)$  as  $t_{k-1} + \tau_k$ , where

$$\tau_k \in \left[ \frac{\gamma_k - \varepsilon}{\gamma_k + \varepsilon} t_d, \frac{\gamma_k + \varepsilon}{\gamma_k - \varepsilon} t_d \right],$$

and  $\gamma(t) = \gamma_k$  is the gain used for  $t_{k-2} + \tau_{k-1} \leq t < t_{k-1} + \tau_k$ . In fact,  $\tau_k$  can be estimated as the mid-point of the interval  $[\frac{\gamma_k - \varepsilon}{\gamma_k + \varepsilon} t_d, \frac{\gamma_k + \varepsilon}{\gamma_k - \varepsilon} t_d]$ , which is

$$\tau_k = \hat{\tau}_k = \frac{\gamma_k^2 + \varepsilon^2}{\gamma_k^2 - \varepsilon^2} t_d.$$

Note that  $\hat{\tau}_k = t_d$  if  $\varepsilon = 0$ .

Thus at the time instant  $t_k + \hat{\tau}_k$ ,  $s(t)$  is predicted to cross zero and to reduce limit cycle amplitude we reduce the gain according to the rule,  $\gamma_{k+1} = \max\{\gamma_k \alpha, \gamma_{\min}\}$ , where  $0 < \alpha < 1$  and  $\gamma_{\min} > \varepsilon$ . The reduction of the gain at the time instants where  $s(t)$  is close to zero is targeted since when  $s(t)$  is away from zero, reducing the gain only serves to extend the transient. Note that the next zero crossing of  $s_m(t)$  may be detected at a time instant  $t_k$  that occurs earlier than  $t_{k-1} + \hat{\tau}_k$ ; in this case,  $\gamma_k$  should still be reduced at  $t_{k-1} + \hat{\tau}_k$  and the detected zero crossing at the time instant  $t_k$  should be ignored.

Note that the gain is not allowed to drop below the value  $\gamma_{\min}$  so that  $\gamma_k > \varepsilon$ . We can prove the following

**Proposition:**  $s(t) \rightarrow [-\gamma_{\min} - \varepsilon, \gamma_{\min} + \varepsilon] \cdot t_d$  as  $t \rightarrow \infty$ .

*Proof:* It can be shown that there exists a time instant,  $t_k$ , such that  $s_m(t_k) = 0$ . Then  $s(t_k - t_d) = 0$ . The proof follows from the claim that if  $\gamma(t) \leq \tilde{\gamma}_k$  for  $t \geq t_k - t_d$  then  $|s(t)| \leq (\varepsilon + \tilde{\gamma}_k) t_d$  for  $t \geq t_k - t_d$ . To prove this claim, we define  $r = s - (\varepsilon + \tilde{\gamma}_k) t_d$  and assume, without loss of generality, that  $r(t) > 0$  for some  $t > t_k - t_d$  while  $r(t_k - t_d) < 0$ . Since  $r$  is continuous, we can find  $\bar{t} \geq t_k - t_d$  such that  $r(\bar{t}) = 0$ ,

$r((\bar{t}, t]) > 0$ . Given that  $\dot{s} = -\gamma(t) \cdot \text{sign}(s(t - t_d)) + w(t)$ ,  $|w(t)| \leq \varepsilon$ , this is only possible if  $s((\bar{t} - t_d, t - t_d]) > 0$ . The latter implies, in turn, that  $\dot{r}((\bar{t}, t]) \leq 0$ , which contradicts  $r((\bar{t}, t]) > 0$ . The proof is complete.  $\square$

Thus the gain reduction procedure guarantees that the limit cycle amplitude can be reduced asymptotically to or below  $(\gamma_{\min} + \varepsilon)t_d$ . Note further that in the above calculation of  $\hat{\gamma}_k$ ,  $\varepsilon$  can be replaced by  $\varepsilon_k = \sup_{t_{k-2} \leq t \leq t_{k-1} + t_k} |w(t)|$  that permits tighter bounding of  $w$  based on the most recent information about  $w(t)$ .

Note that large unmodeled uncertainties, e.g., due to selecting  $\varepsilon$  to bound  $w(t)$  at most but not all conditions, may cause  $s_m(t)$  to not switch by more than the expected time during which it is expected to switch. If no switch in  $s_m(t)$  is detected for more than the threshold time  $n(\frac{\gamma_k + \varepsilon}{\gamma_k - \varepsilon} + 1)t_d$ , where  $n > 1$  then  $\gamma_{k+1}$  should be reinstated to its original, largest value,  $\gamma_0$ . This forces the feedback to respond rapidly to unmodeled uncertainties.

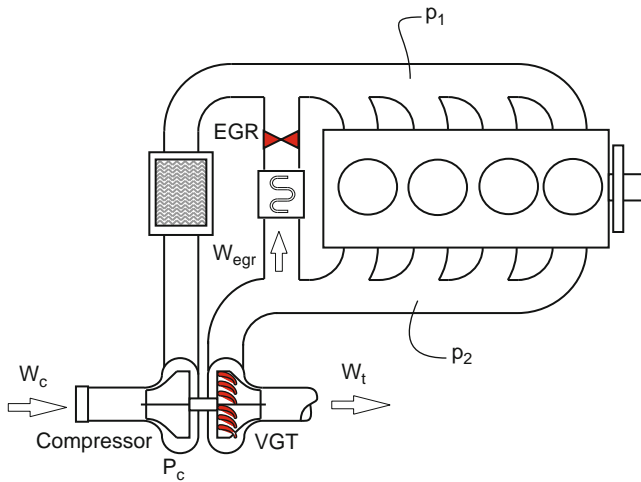
The simulation results for the controller with the adaptive gain are shown in Fig. 3.11 by dash-dotted line. Here  $\gamma_0$  was set to the high gain value ( $= 10^{-3}$ ) of the fixed gain controller and  $\gamma_{\min}$  to the low gain value ( $= 10^{-4}$ ) of the fixed gain controller. The threshold time for increasing the gain was set to  $3\hat{t}_d$  and  $\varepsilon$  to  $1 \times 10^{-4}/2$ . In comparison, the controller with the adaptive gain achieves slightly slower response as compared with the controller with high fixed gain when the air flow changes at  $t = 10.5$  s and it is not correctly estimated. At the same time the amplitude of the limit cycle very quickly settles to much lower value.

### 3.4 Observer Design for a Diesel Engine Model with Time Delay

In this section we consider the problem of observer design for a model of the air-supply system in a diesel engine with an exhaust gas recirculation (EGR) valve and a turbo-compressor with a variable geometry turbine (VGT). The model has a time delay equal to the interval from the air intake into the cylinder to the exhaust of the combusted mixture. Based on the available results, one could argue that the delay does not have a detrimental effect on the performance and robustness of the feedback control system because it can be either dominated [21] or removed from the input-output dynamics by a linearizing feedback [25]. This conclusion applies if all the states of the system are measured. Here we consider the case when only one system's state is available for measurement. We also use the model to discuss issues associated with observer design for systems with time delay.

#### 3.4.1 Diesel Engine Model

A schematic diagram of a diesel engine is shown in Fig. 3.12. The EGR valve is used to recirculate the burned gas from the exhaust into the intake manifold and subsequently into the cylinders in order to reduce NOx emissions. The VGT actuator



**Fig. 3.12** Schematic diagram of a diesel engine with a variable geometry turbine and an exhaust gas recirculation valve.

controls the flow of the exhaust gas through the turbine out of the exhaust manifold. This actuator is employed to improve steady state fuel economy and the system's speed of response. Indeed, fast response is one of the critical objectives of the air-supply system of a diesel engine. A slow response is associated with the driver's perception of a "turbo-lag," the cause of which is the vehicle acceleration being limited by inadequate air supply.

Our starting point is the three-state model of the EGR-VGT diesel engine adopted from [23]. Figure 3.12 shows the compressor air-flow  $W_c$  going through the intercooler (that cools the compressed gas) and filling the intake manifold. The EGR flow  $W_{egr}$ , controlled by the EGR valve, is also cooled by the EGR cooler before entering the intake manifold. The model uses the intake manifold pressure  $p_1$  as one of the system states. The flow of gas from the intake manifold into the engine cylinders is approximately proportional to the manifold pressure. That gas exits the cylinders and enters the exhaust manifold after the "transport" time-delay  $t_d$  approximately equal to  $3/4$  of an engine cycle (from the middle of the intake stroke to the middle of the exhaust stroke). The delay is inversely proportional to engine speed  $N$  and is approximately equal to  $t_d = \frac{90}{N}$  with  $t_d$  in seconds and  $N$  in revolutions per minute. The second state of the system is the exhaust manifold pressure  $p_2$ . While the engine gas flow fills the exhaust manifold, the EGR and turbine flows empty it. The EGR flow from the exhaust manifold into the intake manifold depends on  $p_1$  and  $p_2$  and the opening of the EGR valve. Similarly, the flow through the turbine  $W_t$  depends on exhaust pressure  $p_2$ , the turbine downstream pressure (assumed constant and equal to the ambient), and the position of the VGT actuator. The flow  $W_t$  drives the turbine, that powers the compressor. The third state of the system is, hence, the compressor power  $P_c$ .

The dynamics for the three-state diesel engine model proposed in [23] is given by

$$\begin{aligned}\dot{p}_1 &= k_1(W_c + W_{\text{egr}} - k_e p_1) \\ \dot{p}_2 &= k_2(k_e p_1(t - t_d) - W_{\text{egr}} - W_t + W_f) \\ \dot{P}_c &= \frac{1}{\tau}(\eta_m P_t - P_c),\end{aligned}\tag{3.30}$$

where the intake and exhaust manifold pressures are normalized by the ambient pressure,  $W_f$  is the fuel mass flow rate, and

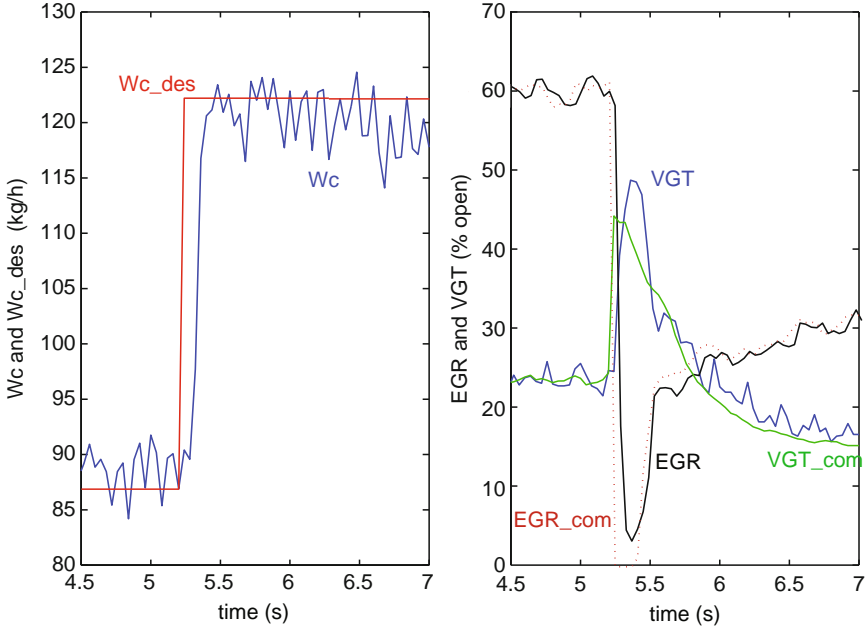
$$W_c = \frac{\eta_c}{T_a c_p} \frac{P_c}{p_1^\mu - 1}, \quad P_t = \eta_t c_p T_2 \left(1 - \frac{1}{p_2^\mu}\right) W_t.\tag{3.31}$$

The coefficients  $\eta_c$ ,  $\eta_t$ , and  $\eta_m$  represent the compressor, turbine, and mechanical efficiencies, respectively,  $c_p$  is the specific heat at constant pressure equal to 1 kJ/(kgK),  $k_e$  is the engine pumping coefficient,  $\tau$  is the time constant of turbine-to-compressor power transfer, and  $k_i = \frac{RT_i}{V_i}$ , with  $V_i$  and  $T_i$ ,  $i = 1, 2$ , being the intake and exhaust manifold volumes and gas temperatures.  $R$  is the specific gas constant for air equal to 0.287 kJ/(kgK). The exponent  $\mu$  is equal to  $\frac{\gamma-1}{\gamma}$ , where  $\gamma$  is the temperature-dependent ratio of specific heats equal to 1.4 at  $T = 300\text{K}$ . The values of  $c_p$  and  $\gamma$  for a range of air temperatures can be found in Appendix D in [17].

The ambient and intake manifold temperatures  $T_a$  and  $T_1$  are assumed constant, while the exhaust temperature  $T_2$  and the parameters  $\eta_t$  and  $\eta_c$  are assumed to depend on the fuel flow rate  $W_f$  and the engine speed  $N$ . For the purpose of control design, the EGR mass flow rate  $W_{\text{egr}}$  and turbine mass flow rate  $W_t$  have been considered the control inputs in [21, 23, 25]. In other words, the desired turbine and EGR flows are generated by the closed-loop controller and the actuator positions are then manipulated to maintain the flows, based on the upstream and downstream pressure signals for each orifice.

Even when the delay is neglected, designing an effective control system for the diesel engine model (3.30) is a difficult task. The system is multivariable and highly nonlinear. For example, it is obvious from the model that, when the normalized pressure  $p_2$  is close to 1, the control  $W_t$  has no effect on compressor power. Moreover, it has been shown in [29] that, due to the nonlinearities, the steady-state input–output gain matrix have entries that change sign within a typical operating region of the engine. Such a DC-gain sign reversal is a major obstacle to using integral action in the controller.

By employing the control Lyapunov function (CLF) concept while assuming  $t_d = 0$ , a proportional multivariable controller was proposed in [23]. The CLF controller, which uses the feedback of the compressor flow  $W_c$  (also referred to as the mass air flow – MAF) and the exhaust pressure  $p_2$ , was compared with several other methods in [41]. The measurement of the intake pressure  $p_1$  was also used by the controller to obtain the EGR valve opening by inverting the EGR flow. The same type of feedback has been shown analytically in [21] to be stabilizing when  $t_d \neq 0$ .



**Fig. 3.13** Experimentally measured response of the compressor mass air flow to a step command (*left*) and the actions of the EGR and VGT actuators (*right*)

The compressor mass air flow signal  $W_c$  is the most relevant, directly measured variable when the air-supply speed of response is considered. Experimental results for this control law, tuned for fast  $W_c$  response, have been reported in [23, 41]. The performance of this full state feedback system<sup>5</sup> is illustrated in Fig. 3.13. From the plot on the left one can estimate the time constant governing the  $W_c$  response to be about 150 ms. The plot on the right shows the percent opening of the EGR and VGT actuators and illustrates the difference between the commanded positions of the actuators (as computed by the feedback controller) and the actual positions attenuated by the actuator dynamics. Despite the noticeable actuator phase lag, the response of  $W_c$  is fast.

The paper [41] also discussed using a reduced set of sensors for the controller. From the cost point of view, it is particularly beneficial to remove the mass air flow sensor for  $W_c$  and use measurements of  $p_1$  and  $p_2$  to estimate it. In the next section, we revisit the sensor selection issue and consider the observer design in the case when only one of the two pressure sensors is available. In particular, we address the issue of observer design in the presence of nonnegligible time-delay  $t_d$ .

<sup>5</sup>  $P_c$  is not directly measured, but can be computed using the first equality in (3.31) from  $W_c$  and  $p_1$  measurements.

### 3.4.2 Observer Design

To preserve the fast system response, it is desirable to place the observer poles to be faster than those of the closed-loop system with full state feedback. This means that the observer time constants need to be close to the value of the time delay, which is equal to 60 ms at  $N = 1,500$  rpm. Thus, the delay may affect the observer performance and robustness more significantly than it did affect the full-state-feedback controller.

When deciding which pressure sensor to use for observer design, there is an interesting trade-off to be considered:

1. Selecting  $y = p_1$  allows direct cancellation of the delay term  $k_2 k_c p_1(t - t_d)$  from the observer error dynamics. This effectively removes the delay as an issue for finding the observer gains that achieve the desired bandwidth. On the other hand,  $p_1$  is a nonminimum phase output with respect to the control input with lower relative degree (EGR). Thus, while the delay effect is avoided, a potential loss of robustness due to the right half plane zero may outweigh the benefits of removing the delay by cancellation.
2. The output  $y = p_2$  is the minimum phase output with respect to both inputs, but the delay term cannot be cancelled and it stays in the dynamics of the observer error. In this case, finding observer gains that achieve the desired performance becomes more difficult.

At this point we select the exhaust pressure as the measured signal,  $y = p_2$ , and design an observer that uses finite time (predictor) integrals to achieve a finite spectrum of the observer error dynamics.

Because the downstream EGR valve pressure  $p_1$  is not measured any more, the first step in the observer design is to express the EGR flow rate  $W_{\text{egr}}$  in the model in terms of  $p_1$ ,  $p_2$ , and the EGR valve flow coefficient  $\alpha(\text{EGR})$ :

$$W_{\text{egr}} = \alpha(\text{EGR}) p_2 \Psi\left(\frac{p_2}{p_1}\right)$$

where  $\Psi(\cdot)$  is the standard subsonic flow correction factor given by equation (2.7) in [23]. The dependence of  $W_{\text{egr}}$  on the upstream temperature  $T_2$  is included in  $\alpha(\cdot)$ . Hence, the model of a diesel engine air supply system takes the form

$$\begin{aligned} \dot{p}_1 &= k_1 \left[ \frac{\eta_c}{T_{acp}} \frac{P_c}{p_1^\mu - 1} + \alpha(\text{EGR}) p_2 \Psi\left(\frac{p_2}{p_1}\right) - k_e p_1 \right] \\ \dot{p}_2 &= k_2 \left[ k_e p_1(t - t_d) - \alpha(\text{EGR}) p_2 \Psi\left(\frac{p_2}{p_1}\right) - W_t + W_f \right] \\ \dot{P}_c &= \frac{1}{\tau} \left[ \eta_m \eta_t c_p T_2 \left( 1 - \frac{1}{p_2^\mu} \right) W_t - P_c \right] \\ y &= p_2. \end{aligned} \tag{3.32}$$

The fuel mass flow rate  $W_f$  is known and the turbine mass flow rate  $W_t$  is a function of known variables:  $p_2$  and the VGT actuator position. Note that only the  $p_1$  state impacts  $p_2$  ( $W_t$  and  $W_f$  are not directly dependent on  $P_c$ ). Thus, that states  $p_1$  and  $P_c$  will have to be estimated based on the impact that  $p_1$  makes on the measured signal  $p_2$  through a nondelay term  $\alpha(\text{EGR})p_2\Psi\left(\frac{p_2}{p_1}\right)$  and a delay term  $k_e p_1(t - t_d)$ . At a typical operating point, magnitude of the impact of the delayed value of  $p_1$  on  $p_2$  will dominate that of the direct (nondelayed) value as we shall see in the numerical example given below.

The next step in the observer design is to linearize the model (3.32) around an equilibrium operating point  $(p_{1e}, p_{2e}, P_{ce})$  (the subscript “e” identifies the value of a variable at the equilibrium point). By defining  $x_1 = p_1 - p_{1e}$ ,  $x_2 = p_2 - p_{2e}$ ,  $x_3 = P_c - P_{ce}$ ,  $u_1 = \alpha(\text{EGR}) - \alpha(\text{EGR}_e)$ ,  $u_2 = W_t - W_{te}$ , and  $d = W_f$ , a known disturbance, we can compute the Jacobian linearization of (3.32) around the equilibrium point and express it in the form:

$$\begin{aligned}\dot{x} &= A_0 x + A_1 x(t - t_d) + B_1 u + B_2 d \\ y &= Cx,\end{aligned}\tag{3.33}$$

where  $x = [x_1 \ x_2 \ x_3]^T$ ,  $u = [u_1 \ u_2]^T$ , and  $C = [0 \ 1 \ 0]$ . In general, the entries of matrices  $A_0$ ,  $A_1$ , and  $B$  depend on the equilibrium point at which the system is linearized. Their form

$$A_0 = \begin{bmatrix} -a_{11} & a_{12} & a_{13} \\ a_{21} & -a_{22} & 0 \\ 0 & a_{32} & -a_{33} \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & 0 & 0 \\ a_{21}^1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} b_{11} & 0 \\ -b_{21} & -b_{22} \\ 0 & b_{32} \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ b_2^2 \\ 0 \end{bmatrix}\tag{3.34}$$

reveals the structure of the system (note: all the coefficients  $a_{ij}$ ,  $b_{ij}$ , as well as  $a_{21}^1$  and  $b_2^2$ , are nonnegative at all equilibrium points).

To estimate the state  $x$  of the system (3.32), we consider the observer of the form

$$\begin{aligned}\dot{\hat{x}} &= A_0 \hat{x} + A_1 \hat{x}(t - t_d) + B_1 u + B_2 d + \sum_{i=0}^l L_i [y(t - it_d) - C\hat{x}(t - it_d)] \\ &\quad + \int_0^{t_d} \Lambda(\theta) [y(t + \theta - lt_d) - C\hat{x}(t + \theta - lt_d)] d\theta,\end{aligned}\tag{3.35}$$

where  $l$  is a positive integer. The dynamics of the observer error  $e = x - \hat{x}$  is, therefore, given by

$$\dot{e} = A_0 e + A_1 e(t - t_d) - \sum_{i=0}^l L_i C e(t - it_d) - \int_0^{t_d} \Lambda(\theta) C e(t + \theta - lt_d) d\theta \tag{3.36}$$

In general, the observer error dynamics will be infinite dimensional. We would like to find values of  $L_i$  and  $\Lambda(\cdot)$  so that (3.36) has a finite set of stable poles  $\pi_j$  that are well damped and are faster than the controller dynamics. Considering the desired (and achievable) speed of response of the closed-loop system, we require

$\text{Re}\{\pi_j\} < -10$  (rad/s). This spectrum assignment problem is solvable if the pair  $(C, A_0 + A_1 e^{-s t_d})$  is spectrally observable [42], that is, if

$$\text{rank} \begin{bmatrix} sI - A_0 - A_1 e^{-s t_d} \\ C \end{bmatrix} = n, \quad \forall s \in \mathcal{C}$$

where  $\mathcal{C}$  denotes the set of complex numbers and  $n$  is the dimension of the system (in this case  $n = 3$ ). One can check that the spectral observability condition is satisfied for the model (3.33) because the (2,1) entries of  $A_0$  and  $A_1$  have the same sign regardless of the operating point.

To find the observer matrices  $L_i$  and  $\Lambda(\cdot)$  that satisfy the requirements we propose to use the duality between the controller and observer design (see, for example, [37] for the controller–observer duality in delay systems). In this way, the observer design problem is transformed into finding a control law for  $v$  that achieves the finite spectrum assignment for the system of the form

$$\dot{z} = A_0^T z + A_1^T z(t - t_d) + C^T v \quad (3.37)$$

This problem can be tackled by several design methods, including the Laplace transform based [8, 33, 43] and the time domain based [22]. In this chapter we shall use the latter because the transformation into the appropriate form is obvious, after which the computation of the control law can follow a well-defined process.

The coordinate transformation consists of reordering the states and splitting them into the strongly controllable part (denoted by  $\xi$ ) and the part that has to be controlled through delay terms (denoted by  $\chi$ ):  $[\chi_1, \chi_2, \xi] = Tz = [z_3, z_1, z_2]$ . Hence, the coordinate transformation matrix  $T$  is given by

$$T = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

In the  $(\chi, \xi)$  coordinates the system takes the form:

$$\begin{aligned} \dot{\chi} &= F\chi + h_0\xi + h_1\xi(t - t_d), \\ \dot{\xi} &= a_{32}\chi_1 + a_{12}\chi_2 - a_{22}\xi + v, \end{aligned} \quad (3.38)$$

where

$$F = \begin{bmatrix} -a_{33} & a_{13} \\ 0 & -a_{11} \end{bmatrix}, \quad h_0 = \begin{bmatrix} 0 \\ a_{21} \end{bmatrix}, \quad h_1 = \begin{bmatrix} 0 \\ a_{21}^1 \end{bmatrix}$$

and the coefficients  $a_{ij}$  correspond to the notation in (3.34). The system (3.38) belongs to the class considered in the “forwarding” section (Sect. 3.2) of [22]. Instead of the more complex Lyapunov-Krasovskii construction of the stabilizing control, here we shall use the simpler spectrum-equivalence result from the same paper. The design proceeds in several steps:

1. Use the control input transformation to remove the  $\chi$ -state terms that affect the  $\xi$ -subsystem:  $v = v_0 - a_{32}\chi_1 - a_{12}\chi_2$ .

2. Replace the delay term  $h_1 \xi(t - t_d)$  in the  $\chi$ -dynamics with the nondelay term  $e^{-F t_d} h_1 \xi(t)$ . After the first two steps, the system (3.38) is converted into the system without time delay:

$$\begin{aligned}\dot{\chi} &= F\chi + (h_0 + e^{-F t_d} h_1) \xi \\ \dot{\xi} &= -a_{22} \xi + v_0\end{aligned}\quad (3.39)$$

3. Find a feedback law

$$v_0 = -K_\chi \chi - k_\xi \xi \quad (3.40)$$

that stabilizes (3.39). Any of the standard control design methods could be used, including pole placement, LQR, and  $H_\infty$ .

4. The gains  $K_\chi = [k_{\chi 1}, k_{\chi 2}]$  and  $k_\xi$  are substituted into the formula ((2.16) in [22])

$$v = -a_{32} \chi_1 - a_{12} \chi_2 - k_\xi \xi - K_\chi \left( \chi + \int_0^{t_d} e^{-F \theta} h_1 \xi(t + \theta - t_d) d\theta \right) \quad (3.41)$$

It has been shown in [22] that the spectrum of the nondelay system (3.39) with the feedback law (3.40) is the same as that of the delay system (3.38) with the feedback law (3.41). Hence the latter has a finite spectrum.

To obtain the observer gains from the above controller we first revert to the original  $z$ -coordinates of the system (3.37):

$$v = -K_0 z - \int_0^{t_d} K_\chi e^{-F \theta} h_1 C z(t + \theta - t_d) d\theta, \quad (3.42)$$

where  $K_0 = [k_{\chi 2} + a_{12}, k_\xi, k_{\chi 1} + a_{32}]$ . Hence, for the observer (3.35),  $l = 1$  and the observer gains  $L_i, i = 0, 1$  and  $\Lambda(\cdot)$  are obtained by transposing the controller gains in (3.42):

$$L_0 = K_0^T, \quad L_1 = 0, \quad \Lambda(\theta) = C^T h_1^T e^{-F^T \theta} K_\chi^T. \quad (3.43)$$

Thus, the observer is given by

$$\begin{aligned}\dot{\hat{x}} &= A_0 \hat{x} + A_1 \hat{x}(t - t_d) + B_1 u + B_2 d + K_0^T (y - C \hat{x}) \\ &\quad + \int_0^{t_d} C^T h_1^T e^{-F^T \theta} K_\chi^T [y(t + \theta - t_d) - C \hat{x}(t + \theta - t_d)] d\theta.\end{aligned}\quad (3.44)$$

The spectrum of the observer error dynamics (3.36) with the gains (3.43) is also finite and is the same as the spectrum of the closed-loop system (3.39) with the control (3.40).

### 3.4.3 Numerical Results

To illustrate the performance of the observer in simulations, we pick operating points for linearization:  $N = 1,500$  (rpm) (hence, the delay is  $t_d = 60$  ms),  $p_{1e} = 1.3$

(bar),  $p_{2e} = 1.5$  (bar), and  $P_c = 1$  (kW). The parameters of the model are selected to represent a generic engine, turbine, and compressor hardware:  $k_c = 26$  (g/s/bar),  $\eta_c = \eta_t = 0.6$ ,  $\eta_m = 0.85$ ,  $\tau = 0.2$  (s). The engine parameters have been assumed independent of the state values. The operating point corresponds to the compressor air flow  $W_c$  of about 26 (g/s) with 30% EGR flow rate. Finally, we have selected the temperature values of  $T_a = 300$  (K),  $T_2 = 700$  (K) and assumed they are independent of the state  $x$ . With typical sizes of the intake and exhaust manifold volumes, the chosen temperatures result in  $k_1 = 0.23$  and  $k_2 = 0.8$ . At this operating point, the matrices that define the dynamics of the linearized system are

$$A_0 = \begin{bmatrix} -27 & 3.6 & 6 \\ 9.6 & -12.5 & 0 \\ 0 & 9 & -5 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 21 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0.26 & 0 \\ -0.9 & -0.8 \\ 0 & 0.18 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0.8 \\ 0 \end{bmatrix}$$

$$C = [0 \ 1 \ 0]. \quad (3.45)$$

We note that our choice to treat  $W_t$ , rather than the position of the VGT actuator, as a control input has resulted in a system that is open loop unstable at this operating point. The instability is not caused by the delay – the matrix  $A_0 + A_1$ , corresponding to the system matrix for  $t_d = 0$ , is unstable with eigenvalues at  $-29$ ,  $-16$ , and  $1$ .

Following the procedure described in the previous subsection, we transform the observer design problem into the controller design for the nondelay system (3.39). This system is given by

$$\begin{aligned} \dot{\chi} &= \begin{bmatrix} -5 & 6 \\ 0 & -27 \end{bmatrix} \chi + \begin{bmatrix} -21 \\ 116 \end{bmatrix} \xi \\ \dot{\xi} &= -12.5 \xi + v_0. \end{aligned} \quad (3.46)$$

We find the controller for this system by employing the LQR method [2] that minimizes the cost

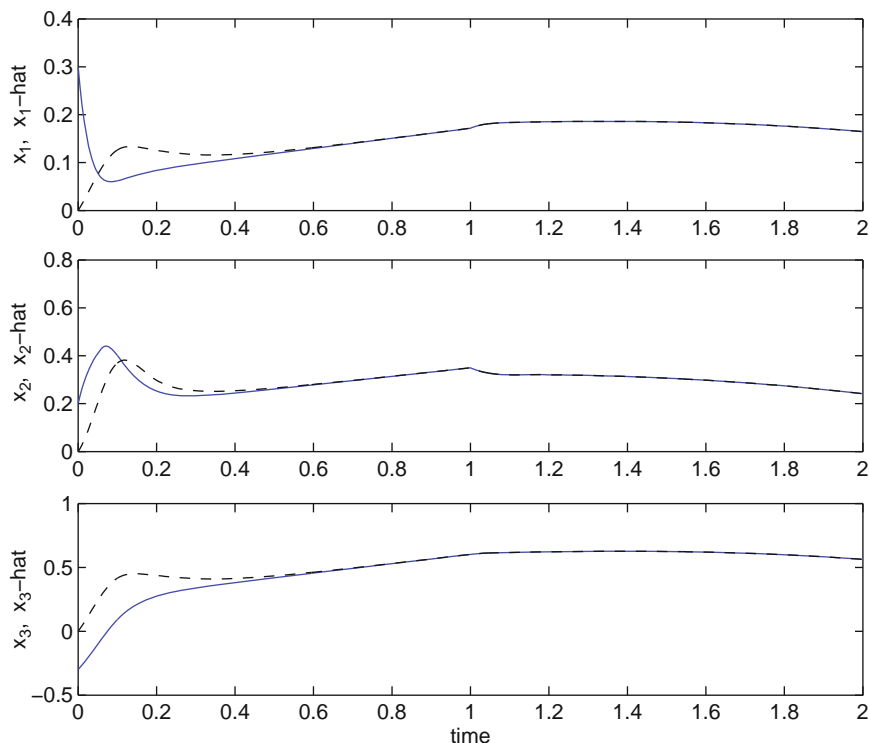
$$J = \int_0^\infty [\chi^T \ \xi] Q \begin{bmatrix} \chi \\ \xi \end{bmatrix} + v_0^2 \, dt.$$

To obtain the poles of the closed-loop system faster than  $-10$  (rad/s), we had to use a nondiagonal matrix  $Q$ . Selecting

$$Q = \begin{bmatrix} 1,000 & 100 & 0 \\ 100 & 15 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

provided the controller gains  $K_\chi = [10.6 \ 3.2]$ ,  $k_\xi = 8.9$ , and the closed-loop poles at  $-16.7 \pm j4.9$  and  $-20$ . The observer gains  $L_0$  and  $\Lambda(\theta)$  are computed from (3.43). The integral in the observer (3.44) is computed using the standard trapezoidal rule with  $\Delta t = 5$  ms.

The performance of the observer is shown in Figure 3.14. The inputs for this simulation run are selected randomly:  $u_1$  is a square wave with amplitude 0.5 and frequency 0.5 (Hz) and  $u_2$  is a sine wave with amplitude 3 and frequency 0.07 (Hz).



**Fig. 3.14** Convergence of the state estimates  $\hat{x}$  (dash curves) to the state  $x$  (solid curves): *top plot* –  $x_1$  and  $\hat{x}_1$ ; *middle plot* –  $x_2$  and  $\hat{x}_2$ ; *bottom plot* –  $x_3$  and  $\hat{x}_3$

### 3.4.4 Summary and Discussion

The control design for the air-supply system in a EGR-VGT turbocharged diesel engine is a difficult problem primarily due to significant system nonlinearities. In this section we have examined the problem of observer design for this system, in which the presence of time delay may also present a problem. The degree of difficulty depends on the selection of the measured output. We have linearized the system and used the controller–observer duality and a recent control design method to find observer gains that satisfy the design objective. The performance is illustrated by numerical simulations.

One can point out several outstanding issues with respect to observer design for a system such as (3.32):

- The observer (3.44) is very easy to design and test in simulations, but an experimental implementation would be somewhat more demanding because of the need to approximately compute the finite time integral.

- An observer design method with partial spectrum assignment, such as [31], does not use finite time integrals and may be easier to implement, provided that the number of undesirable open loop poles is small. On the other hand, computation of open loop poles of an infinite dimensional (open loop) system increases design complexity and may reduce robustness of the resulting observer.
- The authors are not aware of a comparison, or guidelines regarding robustness properties of different observer design methods for systems with time delay.
- As mentioned earlier, the dominant feature of the system (3.32) is significant nonlinearity. Computing Jacobian linearizations at many equilibrium points in the operating region, designing linear observers at each one of them, and implementing the observer by gain scheduling seems prohibitively complex.
- On the other hand, it is not clear if any of the existing observers for nonlinear, time-delay systems, such as the one proposed in [13], are applicable to this problem.

### 3.5 Concluding Remarks

In this chapter, the authors have discussed several application problems in the area of powertrain control from the perspective of control of time-delay systems. The associated models and control problems for ISC and air-to-fuel ratio control in gasoline engines and for state estimation in diesel engines with exhaust gas recirculation and variable geometry turbocharging have been reviewed. Modern engines and transmissions rely on many control functions that are impacted by the delays and in this light we have only covered a small (but representative) subset of relevant application problems.

Nevertheless, the authors' broader point has been that more effective analysis and synthesis methods for time-delay systems can lead to improvements in fuel economy, emissions, and drivability of automotive vehicles. Several connections with the theoretical literature on control of time delay systems have been discussed. It is hoped that this chapter will stimulate further interest in applying new theoretical results to powertrain control and in the development of new theoretical results that address special features of these problems.

**Acknowledgments** V. Averina introduced the authors to the method of [9] for constructing the stability charts, and some of the results in Sect. 3.3 are based on the collaboration with her. G. Song has assisted in obtaining some of the results featured in Sect. 3.3 and S. Magner has conducted experimental measurements of the air-to-fuel ratio delay in the vehicle. The authors also acknowledge A. Annaswamy, S. Diop, T.L. Maizenberg, Y. Orlov, V. Winstead, D. Yanakiev, and Y. Yildiz for past collaborations on the topic of time-delay systems, and D. Hrovat, A. Gibson, J. Michelini, G. Ulsoy, and S. Yi for valuable suggestions that are reflected in the manuscript.

## Appendix: Robust and Stochastic Stability

In this section, we discuss linear matrix inequality (LMI) based techniques for the analysis of robust and stochastic stability, which were used in this chapter.

Consider an uncertain system with a time delay (3.25) and a functional,

$$V = V_1 + V_2,$$

$$V_1 = \left( x(t) + B \int_{t-t_d}^t x(s) ds \right)^T P \left( x(t) + B \int_{t-t_d}^t x(s) ds \right), \quad P > 0,$$

$$V_2 = \int_{t-t_d}^t (s-t+t_d) x^T(s) R x(s) ds, \quad R > 0.$$

After straightforward algebraic manipulations, its time rate of change satisfies

$$\begin{aligned} \dot{V} \leq x^T(t) & \left[ P(A+B+\Delta(t)F) + (A+B+\Delta(t)F)^T P \right. \\ & \left. + t_d(A+B+\Delta(t)F)^T P B R^{-1} B^T P(A+B+\Delta(t)F) + t_d R \right] x(t). \end{aligned}$$

Consequently, a sufficient condition for the asymptotic stability of (3.25) is

$$\begin{aligned} & P(A+B+\Delta(t)F) + (A+B+\Delta(t)F)^T P + t_d R \\ & + t_d(A+B+\Delta(t)F)^T P B R^{-1} B^T P(A+B+\Delta(t)F) < 0, \end{aligned} \quad (3.47)$$

for some  $P > 0$ ,  $R > 0$ , and for all  $t > 0$ .

By using Shur's complement, (3.47) can be replaced by

$$\Theta \triangleq \begin{pmatrix} (A+B+\Delta(t)F)^T P + & t_d B^T P(A+B+\Delta(t)F) \\ P(A+B+\Delta(t)F) + t_d \cdot R & -t_d R \\ t_d(A+B+\Delta(t)F+B)^T P B & \end{pmatrix} < 0. \quad (3.48)$$

Since  $\Delta(t)$  is a scalar, we can rearrange (3.48) as

$$\begin{aligned} \Theta = & \begin{bmatrix} (A+B)^T P + P(A+B) + t_d R & t_d B^T P(A+B) \\ t_d(A+B)^T P B & -t_d R \end{bmatrix} \\ & + \begin{bmatrix} P & t_d B^T P \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta(t)I & 0 \\ 0 & \Delta(t)I \end{bmatrix} \begin{bmatrix} F & 0 \\ 0 & F \end{bmatrix} \\ & + \begin{bmatrix} F^T & 0 \\ 0 & F^T \end{bmatrix} \begin{bmatrix} \Delta(t)I & 0 \\ 0 & \Delta(t)I \end{bmatrix} \begin{bmatrix} P & 0 \\ t_d P B & 0 \end{bmatrix}. \end{aligned} \quad (3.49)$$

Applying Lemma A.5<sup>6</sup> from [7] and using Shur's complement, the following sufficient condition in the form of an LMI is obtained:

<sup>6</sup> In the notations of [7], suppose  $\Gamma^T \Gamma \leq I$ ,  $Y$  is symmetric. Then  $Y + H \Gamma E + E^T \Gamma^T H^T < 0$  if and only if there exists  $\varepsilon > 0$  such that  $Y + \varepsilon H H^T + \frac{1}{\varepsilon} E^T E < 0$ .

$$\begin{pmatrix} (A+B)^T P + P(A+B) + t_d R + \varepsilon F^T F & t_d B^T P(A+B) & P & 0 \\ t_d(A+B)^T P B & -t_d R + \varepsilon F^T F & t_d P B & 0 \\ P & t_d B^T P & -\varepsilon I & 0 \\ 0 & 0 & 0 & -\varepsilon I \end{pmatrix} < 0 \quad (3.50)$$

for some  $P > 0$ ,  $R > 0$ , and  $\varepsilon > 0$ .

Consider next a linear stochastic system with a delay,

$$dx(t) = [Ax(t) + Bx(t - t_d)]dt + Cx(t)dw(t), \quad (3.51)$$

where  $w$  is a scalar standard Wiener process. A sufficient condition ([30], p. 419) for the delay-dependent mean-square stochastic stability of this system is

$$\begin{pmatrix} (A+B)^T P + P(A+B) & t_d(A+B)^T P B \\ +C^T P C + t_d R & -t_d R \\ B^T P(A+B)t_d & -t_d R \end{pmatrix} < 0, \quad (3.52)$$

for  $P > 0$ ,  $R > 0$ .

The software package *SeDuMi* [35] was used in the paper for the numerical solution of the linear matrix inequalities.

## References

1. Alfieri E, Amstutz A, Onder C, and Guzzella L (2007). Automatic design and parameterization of a model-based controller applied to the AF-ratio control of a diesel engine. In: Proceedings of the fifth IFAC Symposium on Advances in Automotive Control, Seascape, Monterey Coast, CA.
2. Anderson B and Moore J (1990). Optimal control: Linear quadratic methods. Prentice-Hall, Englewood Cliffs, NJ.
3. Aquino C (1981). Transient A/F control characteristics of 5-liter central fuel injection engine. SAE Paper 810494.
4. Averina V, Kolmanovsky I, Gibson A, Song G, and Bueler E (2005). Analysis and control of delay-dependent behavior of engine air-to-fuel ratio. In: Proceedings of the 2005 IEEE Conference on Control Applications, Toronto, Canada, pp. 1222–1227.
5. Belkoura L, and Richard J (2006). A distribution framework for the fast identification of linear systems with delays. In: Proceedings of the 6th IFAC Workshop on Time Delay Systems, L'Aquila, Italy.
6. Benga S, Li X, and DeCarlo R (2004). Combined controller–observer design for uncertain time delay systems with applications to engine idle speed control. ASME Journal of Dynamic Systems, Measurement and Control, 126:772–780.
7. Boukas E and Liu Z (2002). Deterministic and stochastic time delay systems. Birkhauser, Boston.
8. Brethé D and Loiseau J (1998). An effective algorithm for finite spectrum assignment of single-input systems with delay. Mathematics and Computers in Simulation 45:339–348.
9. Butcher E, Haitao M, Bueler E, Averina V, and Zsolt S (2004). Stability of linear time-periodic delay-differential equations via Chebyshev polynomials. International Journal for Numerical Methods in Engineering, 59:895–922.

10. Fridman E, Fridman L, and Shustin E (2000). Steady modes in relay control systems with time delay and periodic disturbances. *ASME Journal of Dynamic Systems, Measurement and Control*, 122:732–737.
11. Fridman L (2003). Private communications.
12. Fridman E and Niculescu S (2008). On complete Lyapunov-Krasovskii functional techniques for uncertain systems with fast-varying delays. *International Journal of Robust and Nonlinear Control*, 18(3):364–374.
13. Germani A, Manes C, and Pepe P (2001). An asymptotic state observer for a class of nonlinear delay systems. *Kybernetika*, 37:459–478.
14. Gibson A, Kolmanovsky I, and Hrovat D (2006). Application of disturbance observers to automotive engine idle speed control for fuel economy improvement. In: *Proceedings of the 2006 American Control Conference*, Minneapolis, Minnesota, pp. 1197–1202.
15. Glielmo L, Santini S, and Cascella I (2000). Idle speed control through output feedback stabilization for finite time delay systems. In: *Proceedings of the American Control Conference*, Chicago, Illinois, pp. 45–49.
16. Gomez O, Orlov Y, and Kolmanovsky I (2007). On-line identification of SISO linear time-delay systems from output measurements: Identifiability analysis, identifier synthesis, and application to engine transient fuel identification. *Automatica*, 43(12):2060–2069.
17. Heywood J (1988). *Internal combustion engine fundamentals*. McGraw-Hill, New-York.
18. Hrovat D, Dobbins C, and Powell B (1998). Comments on applications of some new tools to robust stability analysis of spark ignition engine: A case study. *IEEE Transactions on Control Systems Technology*, 6(3):435–436.
19. Hrovat D and Powers W (1998). Computer control systems for automotive powertrains. *IEEE Control Systems Magazine*, August 3–10.
20. Hrovat D and Sun J (1997). Models and control methodologies for IC engine idle speed control design. *Control Engineering Practice*, 5(8):1093–1100.
21. Jankovic M (2001). Control design for a diesel engine model with time delay. In: *Proceedings of the Conference on Decision and Control*, Orlando, Florida.
22. Jankovic M (2007). Forwarding, backstepping, and finite spectrum assignment for time delay systems. In: *Proceedings of the American Control Conference*, New York.
23. Jankovic M, Jankovic M, and Kolmanovsky I (2000). Constructive Lyapunov control design for turbocharged diesel engines. *IEEE Transactions on Control Systems Technology*, 8: 288–299.
24. Jankovic M, Magner S, Hagner D, and Wang Y (2007). Multi-input transient fuel control with auto-calibration. In: *Proceedings of American Control Conference*, New York.
25. Jankovic M, and Kolmanovsky I (1999). Controlling nonlinear systems through time delays: An automotive perspective. In: *Proceedings of 1999 European Control Conference*, Karlsruhe, Germany, paper F1027-1.
26. Khan B and Lehman B (1996). Setpoint PI controllers for systems with large normalized dead-time. *IEEE Transactions on Control Systems Technology*, 4(4):459–466.
27. Kiencke U and Nielsen L (2000). *Automotive control systems for engine, driveline, and vehicle*. Springer, New York.
28. Kolmanovsky I and Yanakiev D (2008). Speed gradient control of nonlinear systems and its applications to automotive engine control. *Journal of SICE, Japan* 47(3).
29. Kolmanovsky I, Moraal P, van Nieuwstadt M, and Stefanopoulou A (1999). Issues in modelling and control of intake flow in variable geometry turbocharged engines. In: *System Modelling and Optimization, Proceedings of 1997 IFIP Conference*, Edited by M. Polis et. al. Chapman and Hall/CRC Research Notes in Mathematics, pp. 436–445.
30. Kolmanovskii V and Myshkis A (1999). *Introduction to the theory and applications of functional differential equations*. Kluwer, Dordrecht.
31. Leyva-Ramos J and Pearson A (1995). An asymptotic modal observer for linear autonomous time lag systems. *IEEE Transactions on Automatic Control*, 40:1291–1294.
32. Magner S (2008). Private communications.
33. Manitius A and Olbrot A (1979). Finite spectrum assignment problem for systems with delays. *IEEE Transactions on Automatic Control*, 24:541–553.

34. Niculescu S and Annaswamy A (2003). An adaptive Smith controller for time delay systems with relative degree  $n^* \leq 2$ . *Systems and Control Letters*, 49(5):347–358.
35. Peaucelle D, Henrion D, and Labit Y (2001). User's guide for SeDuMi interface 1.01: solving LMI problems with SeDuMi.
36. Roduner C, Onder C, and Geering H (1997). Automated design of an air/fuel controller for an SI engine considering the three-way catalytic converter in the  $H_\infty$  approach. In: *Proceedings of the fifth IEEE Mediterranean Conference on Control and Systems*, Paphos, Cyprus.
37. Salamon D (1980). Observers and duality between observation and state feedback for time delay systems. *IEEE Transactions on Automatic Control*, 25:1187–1192.
38. Stotsky A, Egardt B, and Eriksson S (2000). Variable structure control of engine idle speed with estimation of unmeasurable disturbances. *ASME Journal of Dynamic Systems, Measurement, and Control*, 122:599–603.
39. Stotsky A, Kolmanovsky I, and Eriksson S (2004). Composite adaptive and input observer-based approaches to the cylinder flow estimation in spark ignition automotive engines. *International Journal of Adaptive Control and Signal Processing*, 18(2):125–144.
40. Stotsky A, Egardt B, and Eriksson S (2000). Variable structure control of engine idle speed with estimation of unmeasurable disturbances. *ASME Journal of Dynamic Systems, Measurement, and Control*, 122:599–603.
41. van Nieuwstadt M, Kolmanovsky I, Moraal P, Stefanopoulou A, and Jankovic M (2000). Experimental comparison of EGR-VGT control schemes for a high speed diesel engine. *IEEE Control Systems Magazine*, 20:63–79.
42. Watanabe K (1986). Finite spectrum assignment and observer for multivariable systems with commensurate delays. *IEEE Transactions on Automatic Control*, 31:543–550.
43. Watanabe K, Nobuyama E, Kitamori T, and Ito M (1992). A new algorithm for finite spectrum assignment of single-input systems with time delay. *IEEE Transactions on Automatic Control*, 37:1377–1383.
44. Yildiz Y, Annaswamy A, Yanakiev D, and Kolmanovsky I (2007). Adaptive idle speed control for internal combustion engines. In: *Proceedings of the 2007 American Control Conference*, New York, pp. 3700–3705.
45. Yildiz Y, Annaswamy A, Yanakiev D, and Kolmanovsky I (2008). Adaptive air-fuel ratio control for internal combustion engines. In: *Proceedings of the 2008 American Control Conference*, Seattle, pp. 2058–2063.
46. Yildiz Y, Annaswamy A, Yanakiev D, and Kolmanovsky I (2008). Automotive powertrain control problems involving time delay: An adaptive control approach. In: *Proceedings of the ASME 2008 Dynamic Systems and Control Conference*, Ann Arbor, to appear.
47. Yanakiev D (2008). Private communications.
48. Zhang F, Grigoriadis K, Franchek M, and Makki I (2007). Linear parameter-varying lean burn air-fuel ratio control for a spark ignition engine. *ASME Journal of Dynamic Systems, Measurement and Control*, 129:404–414.
49. Winstead V and Kolmanovsky I (2004). Observers for fault detection in networked systems with random delays. In: *Proceedings of American Control Conference*, Boston, pp. 2457–2462.

## Chapter 4

# Stability Analysis and Control of Linear Periodic Delayed Systems Using Chebyshev and Temporal Finite Element Methods

Eric Butcher and Brian Mann

**Abstract** In this chapter, a brief literature review is provided together with detailed descriptions of the authors' work on the stability and control of systems represented by linear time-periodic delay-differential equations using the Chebyshev and temporal finite element analysis (TFEA) techniques. Here, the analysis and examples assume that there is a single fixed discrete delay, which is equal to the principal period. Two Chebyshev-based methods, Chebyshev polynomial expansion and collocation, are developed. After the computational techniques are explained in detail with illustrative examples, the TFEA and Chebyshev collocation techniques are both applied for comparison purposes to determine the stability boundaries of a single degree-of-freedom model of chatter vibrations in the milling process. Subsequently, it is shown how the Chebyshev polynomial expansion method is utilized for both optimal and delayed state feedback control of periodic delayed systems.

**Keywords:** Periodic delay systems · Stability · Temporal finite element analysis · Chebyshev polynomials and collocation · Milling process · Optimal control Delayed state feedback control

### 4.1 Introduction

It has been known for quite some time that many systems in science and engineering can be described by models that include past effects. These systems, where the rate of change in a state is determined by both the past and the present states, are described by delay differential equations (DDEs). Examples from recent literature include applications in robotics, biology, human response time, economics, digital force control, and manufacturing processes [3, 5, 33, 74, 75].

There has been a large amount of recent literature on time-periodic systems with time delay, in both the mathematical and engineering literature. The vast majority of papers in the latter category have been concerned with regenerative chatter in machine tool vibrations. The milling problem, [10, 14, 16, 34, 37, 45, 51–54, 61, 66, 78, 84] in particular, has been the main application in the area of machining processes for this type of mathematical model, namely periodic delay-differential equations (DDEs). Other machining applications have included modulated spindle speed [39, 42, 45, 69] or impedance [67, 68] and honing [26]. These problems have motivated much of the recent work, which has primarily focused on stability [13, 24, 38] and bifurcation analysis [15, 19, 53, 79] of time-periodic delayed systems, as well as the related issue of controller design [18, 50, 83]. Other interesting problems related to response [48], optimization [83], eigenvalue, and parameter identification [55] have also been recently investigated. Our goal here is not to review all of these areas, but to briefly highlight the recent work of the authors in this area during the last few years. This work has been mainly concerned with the stability problem of time-periodic delayed systems, while response and bifurcation analysis as well as problems in delayed feedback control, optimal control, and parameter identification have also received attention. Chebyshev-based methods (polynomial expansion and collocation) and temporal finite element methods are two of the numerical tools that have been used in all of these problems, including applications in machining dynamics (milling and impedance-modulated turning) and the stability of columns with periodic retarded follower forces [49]. The qualitative study of these types of dynamical systems often involves a stability analysis, which is presented in the form of stability charts that show the system stability over a range of parameters [6, 60, 76].

Time-periodic delayed systems (either linear or nonlinear) simultaneously contain two different types of effects in time: namely time-periodic coefficients due to parametric excitation and terms that contain time-delay in which the state is evaluated at a previous time instead of the present time. Although both of these effects have been studied for a long time and have been extensively analyzed *separately* in the literature for many years, the existing literature that concerns their *simultaneous* presence in a system is much smaller and more recent. The majority of researchers who work in this area have therefore had an initial interest in one of them (time delay or parametric excitation) before studying their simultaneous presence in a system. For instance, several researchers have investigated a well-known example of a time-periodic ODE known as Mathieu's equation with its accompanying Strutt-Ince stability chart [59] while others have investigated a simple second order delay equation and its accompanying Hsu-Bhatt-Vyshnegradskii stability chart [32]. Professor Stépán and his students subsequently incorporated parametric excitation into their time-delay systems and produced the first stability charts of the Mathieu equation with time-delay [36] – a prototypical second order system which incorporates both time-periodic coefficients and time delay. In this stability diagram, one can easily see the features of both the Strutt-Ince and Hsu-Bhatt-Vyshnegradskii stability charts.

Mathematically, the main feature of delayed systems and DDEs is that they are infinite dimensional and map initial functions to produce the solution, in contrast to ODEs which are finite-dimensional and map initial values to obtain the solution. Hence, the language and notation of functional analysis is often employed in delayed systems [28, 29, 74]. For linear periodic DDEs, the Floquet theory of linear periodic ODEs can be *partially* extended [27, 77]. Hence, the Floquet transition matrix (or monodromy matrix) whose eigenvalues, the *Floquet multipliers*, determine the stability of periodic ODEs becomes a compact infinite dimensional *monodromy operator* (whose eigenvalues or Floquet multipliers determine the stability) for periodic DDEs. Although the operator theoretically exists in a function space independent of any chosen basis, it can be approximated in a finite number of dimensions as a square matrix (and its eigenvalues computed for stability prediction) only after a certain basis of expansion is chosen, which of course is not unique. Although we use either orthogonal polynomials or collocation points for this basis, many other choices are possible. A larger number of terms included in the basis leads to a larger matrix and hence more eigenvalues and more accuracy. The neglected eigenvalues are generally clustered about the origin due to the compact nature of the operator and hence do not influence the stability if a sufficiently large basis is used to insure convergence. However, although the freedom in choosing the type and size of the basis used implies that the finite matrix approximation to the monodromy operator is not unique, its approximate eigenvalues converge to the largest (in absolute value) exact Floquet multipliers as the size of the basis is increased. Different bases result in different rates of convergence, so that one basis results in a smaller matrix (and hence more efficient computation) than does another basis that results in a larger matrix for the same desired level of accuracy.

The majority of the existing techniques for stability analysis of periodic DDEs are concerned with finding a finite dimensional approximation to the monodromy operator. (Other techniques are concerned with approximating the largest Floquet multipliers without approximating the monodromy operator.) Other than the Chebyshev-based and temporal finite element methods, which we believe to be very competitive for their rates of convergence and ease of use, the main alternative for stability analysis via computation of the approximate monodromy matrix is the semi-discretization method [4, 35, 38, 44, 46]. We will not describe this method or its efficiency here but refer the reader to the above references for more information. We wish to point out the set of MATLAB codes that are applicable to periodic DDEs include the well-known numerical integrator DDE23 [70] and two new suites of codes: PDE-CONT, which does numerical bifurcation and continuation for nonlinear periodic DDEs [79], and DDEC, which yields stability charts and other information for linear periodic DDEs with multiple fixed delays using Chebyshev collocation [11].

## 4.2 Stability of Autonomous and Periodic DDEs

A linear autonomous system of  $n$  DDEs with a single delay can be described in state-space form by

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}_1 \mathbf{x}(t) + \mathbf{A}_2 \mathbf{x}(t - \tau), \\ \mathbf{x}(t) &= \phi(t), \quad -\tau \leq t \leq 0,\end{aligned}\tag{4.1}$$

where  $\mathbf{x}(t)$  is the  $n$ -dimensional state vector,  $\mathbf{A}_1$  and  $\mathbf{A}_2$  are square  $n \times n$  matrices, and  $\tau > 0$ . The characteristic equation for the above system, which is obtained by assuming an exponential solution, becomes

$$|\lambda \mathbf{I} - \mathbf{A}_1 - \mathbf{A}_2 e^{-\lambda \tau}| = 0.\tag{4.2}$$

When compared with the characteristic equation for an autonomous ordinary differential equation (ODE), (4.2) has an infinite number of complex roots, which are the eigenvalues of (4.1). The necessary and sufficient condition for asymptotic stability is that all the roots (called characteristic exponents) must have negative real parts [29].

Another general case to consider is the stability of a time periodic system with a single delay. The general expression for this type of system is

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}_1(t) \mathbf{x}(t) + \mathbf{A}_2(t) \mathbf{x}(t - \tau), \\ \mathbf{x}(t) &= \phi(t), \quad -\tau \leq t \leq 0,\end{aligned}\tag{4.3}$$

where  $T$  is the principal period, i.e.,  $\mathbf{A}_1(t + T) = \mathbf{A}_1(t)$  and  $\mathbf{A}_2(t + T) = \mathbf{A}_2(t)$ . For convenience, we only consider the case where the single fixed delay  $\tau = T$ . However, the following analyses also extend to the case of multiple discrete delays, which are rationally related to each other and to the period  $T$ . Unfortunately, because the system matrices vary with time, it is not possible to obtain a characteristic equation similar to (4.2). Analogous to the time-periodic ODE case, the solution can be written in the form  $\mathbf{x}(t) = \mathbf{p}(t)e^{\lambda t}$ , where  $\mathbf{p}(t) = \mathbf{p}(t + T)$  and  $\lambda_j$  are the characteristic exponents. However, a primary difference exists between the dynamic map of a time periodic ODE and a time periodic DDE, since the monodromy operator  $U$  for the DDE (either autonomous or nonautonomous) is infinite dimensional, while the time-periodic ODE system has a finite dimensional monodromy (Floquet transition) matrix [36].

A discrete solution form for (4.3) that maps the  $n$ -dimensional initial vector function  $\phi(t)$  in the interval  $[-\tau, 0]$  to the state of the system in the first period  $[T - \tau, T]$  (which for the current case where  $T = \tau$  becomes  $[0, \tau]$ ), and subsequently to each period thereafter, can be written in operator form as

$$m_{\mathbf{x}}(i) = U m_{\mathbf{x}}(i - 1),\tag{4.4}$$

where  $m_{\mathbf{x}}$  is an expansion of the solution  $\mathbf{x}(t)$  in some basis during either the current or previous period and  $m_{\mathbf{x}}(0) = m_{\phi}$  represents the expansion of  $\phi(t)$ . Dropping the subscript  $\mathbf{x}$ , the state in the interval  $[0, \tau]$  is thus  $m_1 = U m_{\phi}$ . Thus, the condition

for asymptotic stability requires that the infinite number of characteristic multipliers  $\rho_j = e^{\lambda_j \tau}$ , or eigenvalues of  $U$ , must have a modulus of less than one, which guarantees that the associated characteristic exponents have negative real part.

Since we will study the stability of the eigenvalues (Floquet multipliers) of  $U$ , therefore, it is essential that  $U$  act from a vector space of functions back to the same vector space of functions. Note that if  $\mathbf{A}_2(t) \equiv \mathbf{0}$ , then  $U$  becomes the  $n \times n$  Floquet transition matrix for a periodic ODE system. Thus, in contrast to the ODE case, the periodic DDE system has an infinite number of characteristic multipliers, which must have a modulus of less than one for asymptotic stability. The fact that the monodromy operator is infinite dimensional prohibits a closed-form solution. In spite of this, one can approach this problem from a practical standpoint – by constructing a finite dimensional monodromy matrix  $\mathbf{U}$  that closely approximates the stability characteristics of the infinite dimensional monodromy operator  $U$ . This is the underlying approach that is followed throughout this fourth chapter. Thus, (4.4) becomes  $\mathbf{m}_i = \mathbf{U}\mathbf{m}_{i-1}$ , where  $\mathbf{m}_i$  is a finite-dimensional expansion vector of the state in any desired basis, and only a finite number of characteristic multipliers must be calculated. If a sufficiently large expansion basis is chosen, to insure converge, then the infinite number of neglected multipliers are guaranteed to be clustered about the origin of the complex plane by the compactness of  $U$ , and thus do not influence the stability.

### 4.3 Temporal Finite Element Analysis

A number of prior works have used time finite elements to predict either the stability or temporal evolution of a system [1, 9, 23, 63]. However, temporal finite element analysis was first used to determine the stability of delay equations in reference [6]. The authors examined a second order delay equation that was piecewise continuous with constant coefficients. Following a similar methodology, the approach was adapted to examine second-order delay equations with time-periodic coefficients and piecewise continuity in references [52, 53, 55]. While these prior works were limited to second order DDEs, Mann and Patel [56] developed a more general framework that could be used to determine the stability of DDEs that are in the form of a state space model – thus extending the usefulness of the temporal finite element method to a broader class of systems with time delays.

This section describes the temporal finite element approach of Mann and Patel [56] and applies this technique to a variety of example problems.

#### 4.3.1 Application to a Scalar Autonomous DDE

A distinguishing feature of autonomous systems is that time does not explicitly appear in the governing equations. Some application areas where autonomous DDEs

arise are in robotics, biology, and control using sensor fusion. In an effort to improve the clarity of this section, we first consider the analysis of a scalar DDE before describing the generalized approach. Thus, the stability analysis of the scalar DDE is followed by the analysis of a nonautonomous DDE with multiple states.

Time finite element analysis (TFEA) is a discretization approach that divides the time interval of interest into a finite number of temporal elements. The approach allows the original DDE to be transformed into the form of a discrete map. The asymptotic stability of the system is then analyzed from the characteristic multipliers or eigenvalues of the map. We first consider the following time delay system, originally examined by Hayes [30], which has a single state variable

$$\dot{x}(t) = \alpha x(t) + \beta x(t - \tau), \quad (4.5)$$

where  $\alpha$  and  $\beta$  are scalar parameters and  $\tau = 1$  is the time delay. Since the (4.5) does not have a closed form solution, the first step in the analysis is to consider an approximate solution for the  $j$ th element of the  $n$ th period as a linear combination of polynomials or trial functions. The assumed solution for the state and the delayed state are

$$x_j(t) = \sum_{i=1}^3 a_{ji}^n \phi_i(\sigma), \quad (4.6a)$$

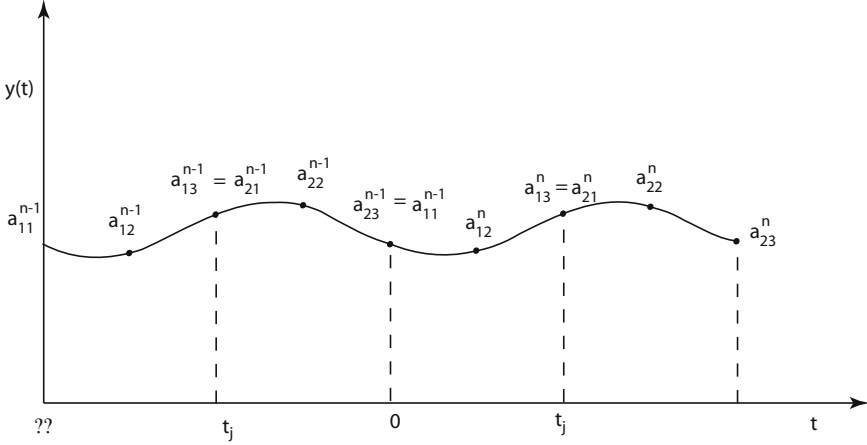
$$x_j(t - \tau) = \sum_{i=1}^3 a_{ji}^{n-1} \phi_i(\sigma), \quad (4.6b)$$

where a superscript is used to denote the  $n$ th and  $n - 1$  delay period for the current and delayed state variable, respectively. Each trial function,  $\phi_i(\sigma)$ , is written as a function of the local time,  $\sigma$ , within the  $j$ th element and the local time is allowed to vary from zero to the time for each element, denoted by  $t_j$ . The introduction of a local time variable is beneficial because it allows the trial functions to remain orthogonal on the interval  $0 \leq \sigma \leq t_j$  once they have been normalized. To further clarify the local time concept, assume that  $E$  elements are used in the analysis and that the time for each element is taken to be uniform, then the time interval for a single element is  $t_j = \tau/E$  and the local time would vary from zero to  $t_j$ . Furthermore, a set of trial functions, orthogonal on the interval from zero to one can be made orthogonal over any  $0 \leq \sigma \leq t_j$  interval by simply replacing the original independent variable of the polynomials with  $\sigma/t_j$ . The polynomials used in all the examples that follow are

$$\phi_1(\sigma) = 1 - 23 \left( \frac{\sigma}{t_j} \right)^2 + 66 \left( \frac{\sigma}{t_j} \right)^3 - 68 \left( \frac{\sigma}{t_j} \right)^4 + 24 \left( \frac{\sigma}{t_j} \right)^5, \quad (4.7a)$$

$$\phi_2(\sigma) = 16 \left( \frac{\sigma}{t_j} \right)^2 - 32 \left( \frac{\sigma}{t_j} \right)^3 + 16 \left( \frac{\sigma}{t_j} \right)^4, \quad (4.7b)$$

$$\phi_3(\sigma) = 7 \left( \frac{\sigma}{t_j} \right)^2 - 34 \left( \frac{\sigma}{t_j} \right)^3 + 52 \left( \frac{\sigma}{t_j} \right)^4 - 24 \left( \frac{\sigma}{t_j} \right)^5. \quad (4.7c)$$



**Fig. 4.1** Timeline for the state variable,  $x$ , over a time interval of  $2\tau$ . Dots denote the locations where the coefficients of the assumed solution are equivalent to the state variable. The beginning and end of each temporal element is marked with *dotted lines*

The above trial functions are orthogonal on the interval of  $0 \leq \sigma \leq t_j$  and they are obtained through interpolation. The interpolated trial functions are constructed such that the coefficients of the assumed solution directly represents the state variable at the beginning  $\sigma = 0$ , middle  $\sigma = t_j/2$ , and end  $\sigma = t_j$  of each temporal element. The graph of Fig. 4.1 is provided to illustrate the important fact that the coefficients of the assumed solution take on the values of the state variables at specific instances in time. Thus, these functions satisfy the natural and essential boundary conditions (i.e., the states at the end of one element match those at the beginning of the following element).

Substituting (4.6a) and (4.6b) into (4.5) results in the following

$$\sum_{i=1}^3 \left( a_{ji}^n \dot{\phi}_i(\sigma) - \alpha a_{ji}^n \phi_i(\sigma) - \beta a_{ji}^{n-1} \phi_i(\sigma) \right) = \text{error}, \quad (4.8)$$

which shows a nonzero error associated with the approximate solutions of (4.6a) and (4.6b). To minimize this error, the assumed solution is weighted by multiplying by a set of test functions, or so called weighting functions, and the integral of the weighted error is set to zero. This is called the method of weighted residuals and requires that the weighting functions be linearly independent [65]. The weighting functions used for the presented analysis were shifted Legendre polynomials. These polynomials were used because they satisfy the required condition of linear independence. Here, we have chosen to only use the first two shifted Legendre polynomials  $\psi_1(\sigma) = 1$  and  $\psi_2(\sigma) = 2(\sigma/t_j) - 1$  to keep the matrices of (4.10) square. The weighted error expression becomes

$$\int_0^{t_j} \left( a_{ji}^n \dot{\phi}_i(\sigma) - \alpha a_{ji}^n \phi_i(\sigma) - \beta a_{ji}^{n-1} \phi_i(\sigma) \right) \psi_p(\sigma) d\sigma = 0. \quad (4.9)$$

After applying each weighting function, a global matrix equation can be obtained by combining the resulting equations for each element. To provide a representative expression, we assume two elements are sufficient and write the global matrix of (4.10). This equation relates the states of the system in the current period to the states of the system in the previous period,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ N_{11}^1 & N_{12}^1 & N_{13}^1 & 0 & 0 \\ N_{21}^1 & N_{22}^1 & N_{23}^1 & 0 & 0 \\ 0 & 0 & N_{11}^2 & N_{12}^2 & N_{13}^2 \\ 0 & 0 & N_{21}^2 & N_{22}^2 & N_{23}^2 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix}^n = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ P_{11}^1 & P_{12}^1 & P_{13}^1 & 0 & 0 \\ P_{21}^1 & P_{22}^1 & P_{23}^1 & 0 & 0 \\ 0 & 0 & P_{11}^2 & P_{12}^2 & P_{13}^2 \\ 0 & 0 & P_{21}^2 & P_{22}^2 & P_{23}^2 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix}^{n-1}. \quad (4.10)$$

The terms inside the matrices of (4.10) are the following scalar terms

$$N_{pi}^j = \int_0^{t_j} \left( \dot{\phi}_i(\sigma) - \alpha \phi_i(\sigma) \right) \psi_p(\sigma) d\sigma, \quad (4.11a)$$

$$P_{pi}^j = \int_0^{t_j} \beta \phi_i(\sigma) \psi_p(\sigma) d\sigma. \quad (4.11b)$$

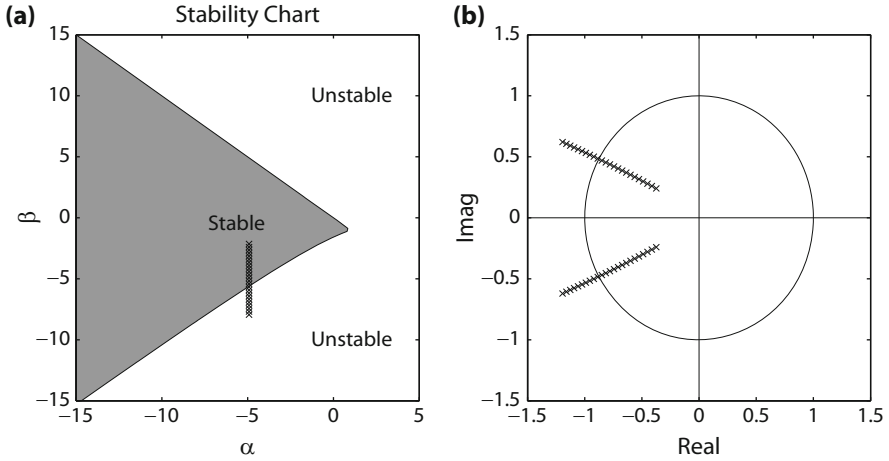
If the time interval for each element is identical, the superscript in these expressions can be dropped for autonomous systems since the expressions for each element would be identical. However, the use of nonuniform time elements or the examination of nonautonomous will typically require the superscript notation.

Equation (4.10) describes a discrete time system or a dynamic map that can be written in a more compact form  $\mathbf{R}\mathbf{a}_n = \mathbf{H}\mathbf{a}_{n-1}$ , where the elements of the  $\mathbf{R}$  matrix are defined by each  $N_{pi}^j$  term of (4.11a). Correspondingly, the elements of the  $\mathbf{H}$  matrix are defined by the  $P_{pi}^j$  terms from (4.11b). Multiplying the dynamic map expression by  $\mathbf{R}^{-1}$  results in  $\mathbf{a}_n = \mathbf{U}\mathbf{a}_{n-1}$ , where  $\mathbf{U} = \mathbf{R}^{-1}\mathbf{H}$ . Applying the conditions of the chosen trial functions to the beginning, midpoint, and end conditions allows us to replace  $\mathbf{a}_n$  and  $\mathbf{a}_{n-1}$  with  $\mathbf{m}_x(n)$  and  $\mathbf{m}_x(n-1)$ , respectively. Here,  $\mathbf{m}_x(n)$  is the vector that represents the variable  $x$  at the beginning, middle, end of each temporal finite element. Thus, the final expression becomes

$$\mathbf{m}_x(n) = \mathbf{U}\mathbf{m}_x(n-1), \quad (4.12)$$

which represents a map of the variable  $x$  over a single delay period (i.e., the  $\mathbf{U}$  matrix relates the state variable at time instances that correspond to the beginning, middle, and end of each element to the state variable one period into the future).

The eigenvalues of the monodromy matrix  $\mathbf{U}$  are called characteristic multipliers. The criteria for asymptotic stability requires that the magnitudes of the characteristic multipliers must be in the modulus of less than one for a given combination of the control parameters. Figure 4.2a shows the boundaries between stable and unstable regions as a function of the control parameters  $\alpha$  and  $\beta$ . The characteristic multipliers trajectories of Fig. 4.2b show how changes in a single control param-



**Fig. 4.2** A converged stability chart (graph (a)) for (4.5) is obtained when using a single temporal element and  $\tau = 1$ . Stable domains are shaded and unstable parameter domains are unshaded. Graph (b) shows the CM trajectories in complex plane for  $\alpha = 4.9$  and a range of values for  $\beta$

eter can cause the characteristic multipliers to exit the unit circle in the complex plane. The authors note that the resulting stability chart is identical those obtained by Kálmar-Nagy [43].

### 4.3.2 TFEA Approach Generalization

Although the scalar case was used provide an introductory example, it is more likely that the practitioner will encounter the case of a first order DDE with multiple states. Thus, this section describes the generalized analysis and its application to some illustrative problems. The general analysis assumes a state space system in the form of (4.3). The expressions for state and the delayed state variables are now written as vectors

$$x_j(t) = \sum_{i=1}^3 \mathbf{a}_{ji}^n \phi_i(\sigma), \quad (4.13a)$$

$$x_j(t - \tau) = \sum_{i=1}^3 \mathbf{a}_{ji}^{n-1} \phi_i(\sigma), \quad (4.13b)$$

during the  $j$ th element. After substituting the assumed solution forms into (4.1) and applying the method of weighted residuals, a global matrix can be obtained that relates the states of the system in the current period to those in the previous period,

$$\begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 \\ \mathbf{N}_{11}^1 & \mathbf{N}_{12}^1 & \mathbf{N}_{13}^1 & 0 & 0 \\ \mathbf{N}_{21}^1 & \mathbf{N}_{22}^1 & \mathbf{N}_{23}^1 & 0 & 0 \\ 0 & 0 & \mathbf{N}_{11}^2 & \mathbf{N}_{12}^2 & \mathbf{N}_{13}^2 \\ 0 & 0 & \mathbf{N}_{21}^2 & \mathbf{N}_{22}^2 & \mathbf{N}_{23}^2 \end{bmatrix} \begin{bmatrix} \mathbf{a}_{11} \\ \mathbf{a}_{12} \\ \mathbf{a}_{21} \\ \mathbf{a}_{22} \\ \mathbf{a}_{23} \end{bmatrix}^n = \begin{bmatrix} 0 & 0 & 0 & 0 & \Phi \\ \mathbf{P}_{11}^1 & \mathbf{P}_{12}^1 & \mathbf{P}_{13}^1 & 0 & 0 \\ \mathbf{P}_{21}^1 & \mathbf{P}_{22}^1 & \mathbf{P}_{23}^1 & 0 & 0 \\ 0 & 0 & \mathbf{P}_{11}^2 & \mathbf{P}_{12}^2 & \mathbf{P}_{13}^2 \\ 0 & 0 & \mathbf{P}_{21}^2 & \mathbf{P}_{22}^2 & \mathbf{P}_{23}^2 \end{bmatrix} \begin{bmatrix} \mathbf{a}_{11} \\ \mathbf{a}_{12} \\ \mathbf{a}_{21} \\ \mathbf{a}_{22} \\ \mathbf{a}_{23} \end{bmatrix}^{n-1}, \quad (4.14)$$

where  $\mathbf{I}$  is an identity matrix and the terms  $\mathbf{N}_{pi}^j$  and  $\mathbf{P}_{pi}^j$  now become the following square matrices

$$\mathbf{N}_{pi}^j = \int_0^{t_j} \left( \mathbf{I} \dot{\phi}_i(\sigma) - \mathbf{A}_1 \phi_i(\sigma) \right) \psi_p(\sigma) d\sigma, \quad (4.15a)$$

$$\mathbf{P}_{pi}^j = \int_0^{t_j} \mathbf{A}_2 \phi_i(\sigma) \psi_p(\sigma) d\sigma. \quad (4.15b)$$

Here, we note the  $\Phi$  matrix is the identity matrix when the delay terms are always present. However,  $\Phi$  may not always be the identity matrix for systems that are piecewise continuous. The dimensions of the matrices  $\mathbf{I}$ ,  $\Phi$ ,  $\mathbf{N}_{pi}^j$ , and  $\mathbf{P}_{pi}^j$  are the same as the dimensions of  $\mathbf{A}_1$  and  $\mathbf{A}_2$ .

The dynamic map of (4.14) is then written in a more compact form,  $\mathbf{R}\mathbf{a}_n = \mathbf{H}\mathbf{a}_{n-1}$ , where the elements of the  $\mathbf{R}$  matrix are defined by each  $\mathbf{N}_{pi}^j$  term of (4.15a). Correspondingly, the elements of the  $\mathbf{H}$  matrix are defined by the  $\mathbf{P}_{pi}^j$  terms from (4.15b). Evaluating the chosen trial functions to the beginning, midpoint, and end of each element allows the coefficients of the assumed solution,  $\mathbf{a}_n$  and  $\mathbf{a}_{n-1}$ , to be replaced with  $\mathbf{m}_x(n)$  and  $\mathbf{m}_x(n-1)$ , respectively. An eigenvalue problem is then formed by multiplying the dynamic map expression by  $\mathbf{R}^{-1}$  and taking the eigenvalues of matrix  $\mathbf{U} = \mathbf{R}^{-1}\mathbf{H}$ . Alternatively, one may wish to avoid the complication of inverting  $\mathbf{R}$ , as in the case that  $\mathbf{R}$  is close to singular, and would instead prefer to solve for the  $\rho$  values (the characteristic multipliers) that solve the characteristic equation that is obtained by setting the determinant of  $\mathbf{H} - \rho\mathbf{R}$  equal to zero.

### 4.3.3 Application to Time-Periodic DDEs

In this section, we consider the Mathieu equation as the case of a damped and delayed oscillator. The damped delayed Mathieu equation (DDME) provides as representative system with the combined effect of parametric excitation and a single time delay. The original version of Mathieu's Equation did not contain either damping or a time delay and was discussed first in 1868 by Mathieu [58] to study the vibration of an elliptical membrane. Bellman and Cook [8] and Hsu and Bhatt [32] both made attempts to lay out the criteria for stability using D-subdivision method combined with the theorem of Pontryagin [64]. Insperger and Stépán used analytical and semidiscretization approach in references [35, 36, 38] and Garg et al. [24] used a second order temporal finite element analysis to investigate the stability of the DDME. The equation of interest is

$$\ddot{x}(t) + \kappa \dot{x}(t) + \left( \delta + \varepsilon \cos(\omega t) \right) x(t) = b x(t - \tau), \quad (4.16)$$

where the equation has a period of  $T = 2\pi/\omega$ , a damping coefficient of  $\kappa$ , a constant time delay  $\tau=2\pi$ . The parameter  $b$  acts much like the gain in a state variable feedback system to scale the influence of the delayed term. For the results of this section, the parameter  $\omega$  is set to one. According to the extended Floquet theory for DDEs, this requires the monodromy matrix to be constructed over the period of the time-periodic matrices  $T = 2\pi/\omega = 2\pi$ .

The first step in the analysis is to rewrite (4.16) as a state space equation,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -(\delta + \varepsilon \cos(\omega t)) & -\kappa \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ b & 0 \end{bmatrix} \begin{bmatrix} x_1(t - \tau) \\ x_2(t - \tau) \end{bmatrix}, \quad (4.17)$$

where  $y_1 = x$  and  $y_2 = \dot{x}$ . Once the equation is written in the form of a state space model, it becomes apparent that the more generalized form is (4.3). This nonautonomous case has two matrices,  $\mathbf{A}_1(t)$  and  $\mathbf{A}_2$ , which are given by

$$\mathbf{A}_1(t) = \begin{bmatrix} 0 & 1 \\ -\delta - \varepsilon \cos(\omega t) & -\kappa \end{bmatrix}, \quad \text{and} \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 0 \\ b & 0 \end{bmatrix}. \quad (4.18)$$

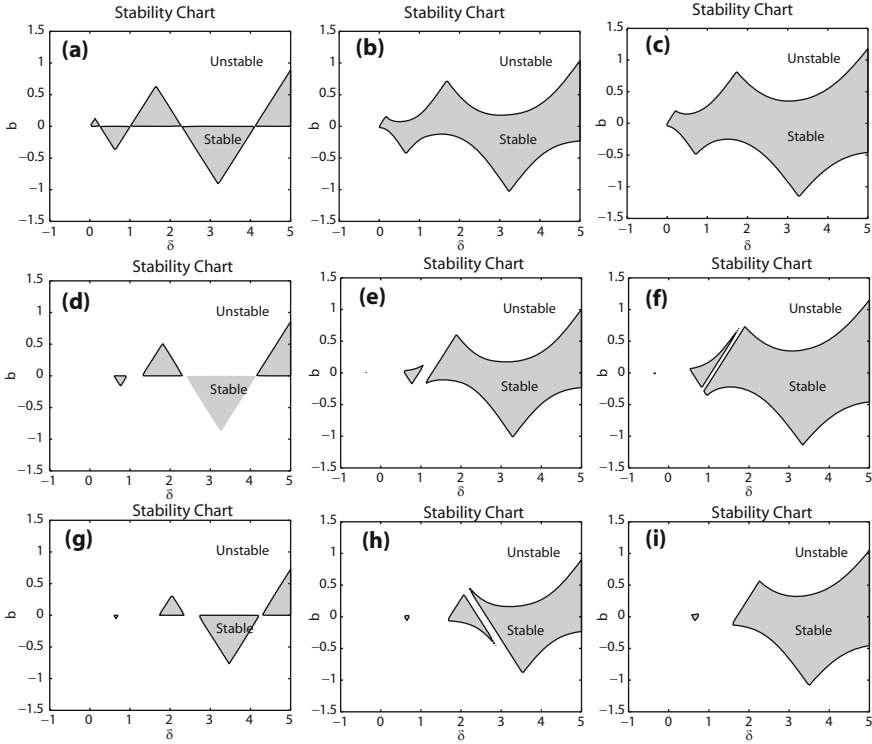
Once again, the solution process starts by substituting (4.13a) and (4.13b) into (4.3). The solution for the  $j$ th element then requires a slight alteration to the time-periodic terms inside the matrices. Assuming that  $E$  uniform temporal elements are applied, the time duration for each element would be  $t_j = T/E$ . Next, we substitute  $t = \sigma + (j-1)t_j$  into the matrix  $\mathbf{A}_1(t)$  so that the cosine term takes on the correct values over the entire period  $T = 2\pi/\omega$ . These terms are then substituted into (4.3) and the method of weighted residuals is applied – as in the previous sections. The expressions that populate the matrix of (4.14) are

$$\mathbf{N}_{pi}^j = \int_0^{t_j} \left( \mathbf{I} \dot{\phi}_i(\sigma) - \mathbf{A}_1(\sigma + (j-1)t_j) \phi_i(\sigma) \right) \psi_p(\sigma) d\sigma, \quad (4.19a)$$

$$\mathbf{P}_{pi}^j = \int_0^{t_j} \mathbf{A}_2 \phi_i(\sigma) \psi_p(\sigma) d\sigma, \quad (4.19b)$$

and  $\Phi$  is taken to be the identity matrix. Here, we point out that the superscript,  $j$ , may not be dropped in the above expressions since the time-periodic terms of  $\mathbf{A}_1(\sigma + (j-1)t_j)$  will assume different values within each element.

Figure 4.3 shows a series of stability charts for  $\omega = 1$ ,  $\tau = 2\pi$ , and different values of  $\varepsilon$  and  $\kappa$ . While the presented results are for the case of  $T$  being equivalent to the time delay, additional cases, – such as those with a nonequivalent time interval, can be found in [24, 38]. The stability charts of Fig. 4.3 show that as the damping is increased the stable parameter space grows. It can also be observed that the stable parameter space begins to unify as the damping term is increased. Finally, when the amplitude of the parametric excitation is increased to larger values of  $\varepsilon$ , the stability regions again become disjoint.



**Fig. 4.3** Stability chart for (4.16) using  $\omega = 1$  and  $\tau = 2\pi$  and five elements. Stability results for each graph are for the following parameters: (a)  $\varepsilon = 0$  and  $\kappa = 0$ ; (b)  $\varepsilon = 0$  and  $\kappa = 0.1$ ; (c)  $\varepsilon = 0$  and  $\kappa = 0.2$ ; (d)  $\varepsilon = 1$  and  $\kappa = 0$ ; (e)  $\varepsilon = 1$  and  $\kappa = 0.1$ ; (f)  $\varepsilon = 1$  and  $\kappa = 0.2$ ; (g)  $\varepsilon = 2$  and  $\kappa = 0$ ; (h)  $\varepsilon = 2$  and  $\kappa = 0.1$ ; (i)  $\varepsilon = 2$  and  $\kappa = 0.2$

## 4.4 Chebyshev Polynomial Expansion and Collocation

The use of Chebyshev polynomials to obtain the solutions of differential equations dates back to the works of Clenshaw [17], Elliot [20], and Wright [82] and was summarized in the books by Fox and Parker [22] and Snyder [73]. As stated in [22], a simple power series solution is not the best convergent solution on a finite interval. Instead, if the solution is expressed in terms of Chebyshev polynomials, higher accuracy and better convergence are achieved. Although other orthogonal polynomials have also been used, Chebyshev polynomials are optimal in terms of minimizing the uniform error over the entire interval [73]. Although in these early works recurrence relations are explicitly used, in later vector/matrix formulations various operational matrices associated with the Chebyshev polynomials (or their shifted versions) are utilized instead. Such formulations were developed for several specific problems,

including the response and stability analysis of linear time-periodic ODEs [71] and the response of constant-coefficient DDEs [31].

#### 4.4.1 Expansion in Chebyshev Polynomials

The standard Chebyshev polynomials are defined as [22]:

$$T_r(x) = \cos r\theta, \quad \cos \theta = x, \quad -1 \leq x \leq 1 \quad (4.20)$$

Using the change of variable  $x = 2t - 1$ ,  $0 \leq x \leq 1$ , the shifted Chebyshev polynomials are defined in the interval  $t \in [0, 1]$  as

$$T_r^*(x) = T_r(2t - 1) \quad (4.21)$$

Note  $|T_r^*(t)| \leq 1$ . Suppose  $f(t)$  is a continuous scalar function, which can be expanded in shifted Chebyshev polynomials:

$$f(t) = \sum_{j=0}^{\infty} b_j T_j^*(t), \quad 0 \leq t \leq 1. \quad (4.22)$$

Using the orthogonality property of the polynomials, the coefficients  $b_j$  are given by

$$b_j = \frac{2}{\pi} \int_0^1 f(t) T_j^*(t) w(t) dt, \quad j = 1, 2, 3, \dots, \quad (4.23)$$

$$b_0 = \frac{1}{2} \int_0^1 f(t) w(t) dt, \quad j = 0,$$

where  $w(t) = (t - t^2)^{-1/2}$ . Our notation for a finite expansion in the first  $m$  shifted Chebyshev polynomials is

$$f(t) = \sum_{j=0}^{m-1} a_j T_j^*(t) = \mathbf{T}(t)^T \mathbf{a}, \quad (4.24)$$

where  $\mathbf{T}(t) = \{T_0^*(t) T_1^*(t) \dots T_{m-1}^*(t)\}^T$  and  $\mathbf{a}$  are  $m \times 1$  column vectors of the polynomials and coefficients, respectively. Linear operations on functions can now be written as matrix operations on vectors of polynomials and coefficients, respectively. To build a square monodromy matrix whose eigenvalues will determine the stability of the DDE, we employ square matrix approximations to these operations.

Integration can be represented with a small error by a square matrix [2]:

$$\int_0^t \mathbf{T}(\tau) d\tau = \mathbf{G} \mathbf{T}(t) + O(m^{-1}), \quad (4.25)$$

where  $\mathbf{G} =$

$$\begin{bmatrix} 1/2 & 1/2 & 0 & 0 & 0 & \cdots & 0 \\ -1/8 & 0 & 1/8 & 0 & 0 & \cdots & 0 \\ -1/6 & -1/4 & 0 & 1/12 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 1/4(m-1) \\ (-1)^m/2m(m-2) & 0 & 0 & \cdots & 0 & 1/4(m-2) & 0 \end{bmatrix} \quad (4.26)$$

is an  $m \times m$  integration operational matrix. Using the relation  $T_r^*(t)T_k^*(t) = (T_{r+k}^*(t) + T_{|r-k|}^*(t))/2$  for the product of two shifted Chebyshev polynomials, the operation of multiplying two Chebyshev-expanded scalar functions  $f(t) = \mathbf{T}(t)^T \mathbf{a}$  and  $g(t) = \mathbf{T}(t)^T \mathbf{b}$  is approximated as

$$f(t)g(t) = \mathbf{T}(t)^T \mathbf{Q}_a \mathbf{b}, \quad (4.27)$$

using the square  $m \times m$  product operational matrix

$$\mathbf{Q}_a = \begin{bmatrix} a_0 & a_1/2 & a_2/2 & a_3/2 & \cdots & a_{m-1}/2 \\ a_1 & a_0 + a_2/2 & 1/2(a_1 + a_3) & 1/2(a_2 + a_4) & \cdots & a_{m-2}/2 \\ a_2 & 1/2(a_1 + a_3) & a_0 + a_4/2 & 1/2(a_1 + a_5) & \cdots & a_{m-3}/2 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m-1} & a_{m-2}/2 & a_{m-3}/2 & a_{m-4}/2 & \cdots & a_0 \end{bmatrix} \quad (4.28)$$

This approximation involves dropping  $m(m-1)/2$  terms from the product. If  $f, g$  are at least twice differentiable then the order of the total error in using  $\mathbf{Q}_a$  goes to zero as  $m$  increases [13].

Suppose  $\mathbf{A}(t)$  is any  $n \times n$  matrix-valued function (such as  $\mathbf{A}_1(t)$  or  $\mathbf{A}_2(t)$  in (4.3)) whose entries  $a_{ij}(t)$  can be expanded in  $m$  shifted Chebyshev polynomials. Then

$$\mathbf{A}(t) = \hat{\mathbf{T}}(t)^T \bar{\mathbf{A}} = (\mathbf{I}_n \otimes \mathbf{T}(t)^T) \begin{bmatrix} \mathbf{a}_{11} & \cdots & \mathbf{a}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{a}_{n1} & \cdots & \mathbf{a}_{nn} \end{bmatrix}, \quad (4.29)$$

where  $\mathbf{I}_n$  is the  $n \times n$  identity,  $\otimes$  is the Kronecker product, and  $\mathbf{a}_{ij}$  is an  $m \times 1$  vector of the coefficients of the matrix entry  $A_{ij}(t)$ . Therefore, the integral operation is written

$$\int_0^t \mathbf{A}(\tau) d\tau = \hat{\mathbf{T}}(t)^T \hat{\mathbf{G}}^T \bar{\mathbf{A}}, \quad (4.30)$$

where  $\hat{\mathbf{G}} = \mathbf{I}_n \otimes \mathbf{G}$  is an  $nm \times nm$  matrix. Suppose  $\mathbf{B}(t) = \hat{\mathbf{T}}(t)^T \bar{\mathbf{B}}$  is another matrix function. Then the product is written

$$\mathbf{A}(t)\mathbf{B}(t) = \hat{\mathbf{T}}(t)^T \hat{\mathbf{Q}}_A \bar{\mathbf{B}}, \quad (4.31)$$

where

$$\hat{\mathbf{Q}}_A = \begin{bmatrix} \mathbf{Q}_{a_{11}} & \cdots & \mathbf{Q}_{a_{1n}} \\ \vdots & \ddots & \vdots \\ \mathbf{Q}_{a_{n1}} & \cdots & \mathbf{Q}_{a_{nn}} \end{bmatrix} \quad (4.32)$$

is an  $nm \times nm$  matrix.

Now consider (4.3) in the special case where  $T = \tau$ . Here we present a “direct” formulation for the approximate  $\mathbf{U}$  matrix using Chebyshev polynomials. By integrating once as

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \int_{t_0}^t (\mathbf{A}_1(s)\mathbf{x}(s) + \mathbf{A}_2(s)\mathbf{x}(s - \tau)) ds, \quad (4.33)$$

and normalizing the period to  $T = \tau = 1$ , we obtain the solution vector  $\mathbf{x}_1(t)$  in the first interval  $[0, 1]$  as

$$\mathbf{x}_1(t) = \mathbf{x}_1(0) + \int_0^t (\mathbf{A}_1(s)x(s) + \mathbf{A}_2(s)\phi(s - 1)) ds. \quad (4.34)$$

Next, we expand  $\mathbf{x}_1(t)$ ,  $\mathbf{A}_1(t)$ , and  $\mathbf{A}_2(t)$  and the initial function  $\phi(t - 1)$  in shifted Chebyshev polynomials as

$$\begin{aligned} \mathbf{x}_1(t) &= \hat{\mathbf{T}}(t)^T \mathbf{m}_1, & \mathbf{A}_1(t) &= \hat{\mathbf{T}}(t)^T \bar{\mathbf{A}}_1, & \mathbf{A}_2(t) &= \hat{\mathbf{T}}(t)^T \bar{\mathbf{A}}_2, \\ \phi(t - 1) &= \hat{\mathbf{T}}(t)^T \mathbf{m}_0 & \mathbf{x}(0) &= \hat{\mathbf{T}}(t)^T \bar{\mathbf{T}}(1) \mathbf{m}_0, \end{aligned} \quad (4.35)$$

where  $\mathbf{m}_1$  and  $\mathbf{m}_0$  are the  $nm \times 1$  Chebyshev coefficients vectors of the solution vector  $\mathbf{x}_1(t)$  and the initial function  $\phi(t - 1)$ . The  $nm \times nm$  matrix  $\bar{\mathbf{T}}(1)$  is defined as  $\bar{\mathbf{T}}(1) = \hat{\mathbf{I}}\hat{\mathbf{T}}(1)^T$  where  $\mathbf{I} = \hat{\mathbf{T}}(t)^T \hat{\mathbf{T}}$ . Using the Chebyshev expansions in (4.35), (4.34) takes the form

$$\hat{\mathbf{T}}(t)^T \mathbf{m}_1 = \hat{\mathbf{T}}(t)^T \bar{\mathbf{T}}(1) \mathbf{m}_0 + \int_0^t (\hat{\mathbf{T}}(s)^T \bar{\mathbf{A}}_1 \hat{\mathbf{T}}(s)^T \mathbf{m}_1 + \hat{\mathbf{T}}(s)^T \bar{\mathbf{A}}_2 \hat{\mathbf{T}}(s)^T \mathbf{m}_0) ds. \quad (4.36)$$

Applying the operational matrices and simplifying, we obtain

$$\hat{\mathbf{T}}(t)^T [\mathbf{I} - \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_1}] \mathbf{m}_1 = \hat{\mathbf{T}}(t)^T [\bar{\mathbf{T}}(1) + \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_2}] \mathbf{m}_0. \quad (4.37)$$

For the  $i$ th interval  $[i-1, i]$ , the linear map  $\mathbf{U}$  which relates the Chebyshev coefficient vector  $\mathbf{m}_i$  to that in the previous interval is therefore given by

$$\mathbf{m}_i = [\mathbf{I} - \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_1}]^{-1} [\bar{\mathbf{T}}(1) + \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_2}] \mathbf{m}_{i-1}, \quad (4.38)$$

which is equivalent to (4.4). Hence, the stability matrix  $\mathbf{U}$  is the linear map in (4.38). The matrix  $\mathbf{U}$  can be considered as a finite-dimensional approximation to the infinite-dimensional operator, which maps continuous functions from the interval  $[0, 1]$  back to the same interval. For asymptotic stability, all the eigenvalues  $\rho_i$  of  $\mathbf{U}$  must lie within, or, for the neutral stability, on the unit circle. Alternatively, the inversion of  $[\mathbf{I} - \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_1}]$  can be avoided by setting the determinant of  $\{[\bar{\mathbf{T}}(1) + \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_2}] - \rho[\mathbf{I} - \hat{\mathbf{G}}^T \hat{\mathbf{Q}}_{A_1}]\}$  to zero.

#### 4.4.2 Estimating the Number of Polynomials

Since one can determine the exponential growth rate of a system from easily calculated norms of the coefficient matrices, from this information one can choose a sufficient value for the important numerical parameter  $m$  (number of shifted Chebyshev polynomials) to give a desired accuracy in a particular example. Suppose that  $p(t)$  is the best shifted Chebyshev polynomial expansion of  $f(t)$  using only the  $m$  polynomials  $T_0^*(t), \dots, T_{m-1}^*(t)$ . We depend on the following relationship which relates the accuracy of the Chebyshev approximation  $p$  to the size of the  $m$  continuous derivatives of  $f$ :

$$\max_{0 \leq t \leq 1} |f(t) - p(t)| \leq \frac{1}{2^{2m-1} m!} \max_{0 \leq t \leq 1} |f^{(m)}(t)|. \quad (4.39)$$

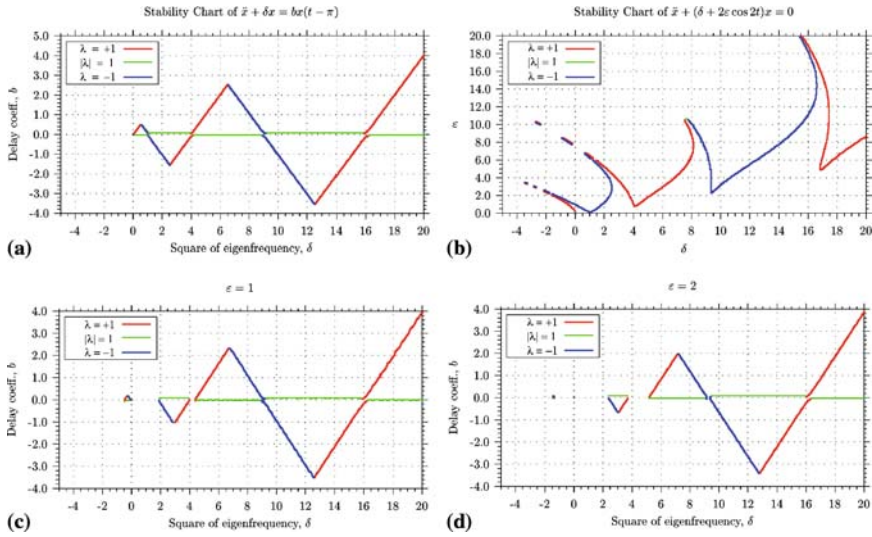
Suppose we want to apply these methods to the scalar constant coefficient equation

$$\dot{x} = 3x - 2x(t-1), \quad (4.40)$$

and with  $x(s) = \phi(s)$ ,  $-1 \leq s \leq 0$ . Suppose we want to have an error of at most  $10^{-6}$ . Since the solution is a function with exponential growth at most  $e^{3t}$ , from (4.40) we need  $3^m e^3 / (2^{2m-1} m!) < 10^{-6}$ , that is,  $m \geq 10$ . The heuristic  $x(t) \sim e^{\bar{\rho}t}$  along with (4.39) can be used in the general case, where

$$\bar{\rho} = \max_{0 \leq s \leq 1} \rho(\mathbf{A}_1(s)), \quad (4.41)$$

is the maximum spectral radius over the normalized period.



**Fig. 4.4** The stability diagrams of the delayed Mathieu equation (4.42) with (a)  $\varepsilon = 0$ , (b)  $b = 0$ , (c)  $\varepsilon = 1$ , and (d)  $\varepsilon = 2$ . Reprinted from *International Journal for Numerical Methods in Engineering*, vol. 59, pp. 895-922, “Stability of linear time-periodic delay-differential equation via Chebyshev polynomials” by E. A. Butcher, et al., 2004. ©John Wiley & Sons Limited. Reproduced with permission

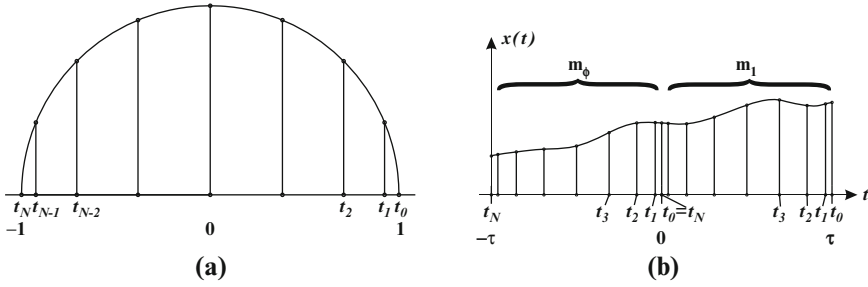
For example, for the delayed Mathieu equation [36],

$$\ddot{x}(t) + (\delta + 2\varepsilon \cos 2t)x(t) = bx(t - \pi) \quad (4.42)$$

after normalizing, we suppose that the solution has approximate exponential form  $e^{\bar{\rho}t}$  where  $\bar{\rho} = \pi\sqrt{|\delta| + 2|\varepsilon|}$ . (We are disregarding the parameter  $b$  because it only contributes a subexponential term). Suppose, as in Fig. 4.4, that we wish to determine stability for parameter ranges  $-5 \leq \delta \leq 20$ ,  $-4 \leq b \leq 4$ ,  $0 \leq \varepsilon \leq 20$  with an error of at most  $10^{-6}$ . Then  $\bar{\rho} = \pi\sqrt{60} \approx 24.3$  is the worst case and we seek  $m$  so that  $(\bar{\rho}^m e^{\bar{\rho}})/(2^{2m-1}m!) \leq 10^{-6}$ , that is,  $m \geq 41$ . We consider four special cases of the delayed Mathieu equation (4.42) to plot in Figure 4.4 using Chebyshev polynomial expansion. These include (a)  $\varepsilon = 0$  (first obtained in [32]); (b)  $b = 0$  (first obtained in [81]); (c)  $\varepsilon = 1$ ; and (d)  $\varepsilon = 2$  (first obtained in [36]).

### 4.4.3 Chebyshev Collocation Method

We now illustrate a different Chebyshev-based method, which uses a collocation expansion of the solution at the extrema of the Chebyshev polynomials. In fact, the collocation method is more efficient than the method of polynomial expansion. Unlike that method, it can easily be applied to DDEs with nonsmooth coefficients. It



**Fig. 4.5** Diagrams of (a) Chebyshev collocation points as defined by projections from the unit circle and (b) collocation vectors on successive intervals

generalizes and extends the collocation and pseudospectral techniques for constant-coefficient DDEs and boundary value problems introduced in [7, 40, 80] to approximate the compact monodromy operator of a periodic DDE, whose eigenvalues exponentially converge to the exact Floquet multipliers. It is flexible for systems with multiple degrees of freedom and it produces stability charts with high speed and accuracy in a given parameter range. It should be noted that, unlike in [21] in which collocation methods were used to find periodic solutions to nonlinear DDEs, the method proposed here explicitly computes stability properties of linear periodic DDEs which may have been obtained through linearization about a periodic solution. An *a priori* proof of convergence was sketched in [25], and computable uniform *a posteriori* bounds were given for this method in [12].

The Chebyshev collocation method is based on the properties of the Chebyshev polynomials. The Chebyshev collocation points are unevenly spaced points in the domain  $[-1, 1]$  corresponding to the extremum points of the Chebyshev polynomial. As seen in Fig. 4.5a, we can also define these points as the projections of equispaced points on the upper half of the unit circle as  $t_j = \cos(j\pi/N)$ ,  $j = 0, 1, \dots, N$ . Hence, the number of collocation points used is  $m = N + 1$ . A spectral differentiation matrix for the Chebyshev collocation points is obtained by interpolating a polynomial through the collocation points, differentiating that polynomial, and then evaluating the resulting polynomial at the collocation points [12]. We can find the differentiation matrix  $\mathbf{D}$  for any order  $m$  as follows: Let the rows and columns of the  $m \times m$  Chebyshev spectral differentiation matrix  $\mathbf{D}$  be indexed from 0 to  $N$ . The entries of this matrix are

$$D_{00} = \frac{2N^2 + 1}{6}, D_{NN} = -\frac{2N^2 + 1}{6}, D_{jj} = \frac{-t_j}{2(1 - t_j^2)}, j = 1, \dots, N-1$$

$$D_{ij} = \frac{c_i(-1)^{i+j}}{c_i(t_i - t_j)}, i \neq j, i, j = 0, \dots, N, c_i = \begin{cases} 2, & i = 0, N \\ 1, & \text{otherwise} \end{cases} \quad (4.43)$$

The dimension of  $\mathbf{D}$  is  $m \times m$ . Also let the  $mn \times mn$  differential operator  $\mathbb{D}$  be defined as  $\mathbb{D} = \mathbf{D} \otimes I_n$ .

Now let us approximate (4.3) using the Chebyshev collocation method, in which the approximate solution is defined by the function values at the collocation points in any given interval. (Note that for a collocation expansion on an interval of length  $T = \tau$ , the standard interval  $[-1, 1]$  for the Chebyshev polynomials is easily rescaled). As shown in Fig. 4.5b, let  $\mathbf{m}_1$  be the set of  $m$  values of  $\mathbf{x}(t)$  in the interval  $t \in [0, T]$  and  $\mathbf{m}_\phi$  be the set of  $m$  values of the initial function  $\phi(t)$  in  $t \in [-T, 0]$ . Recalling that the points are numbered right to left by convention, the matching condition in Fig. 4.5b is seen to be that  $\mathbf{m}_{1N} = \mathbf{m}_{\phi 0}$ . Writing (4.3) in the algebraic form representing the Chebyshev collocation expansion vectors  $\mathbf{m}_\phi$  and  $\mathbf{m}_1$ , we obtain

$$\hat{\mathbf{D}}\mathbf{m}_1 = \hat{\mathbf{M}}_{A_1}\mathbf{m}_1 + \hat{\mathbf{M}}_{A_2}\mathbf{m}_\phi. \quad (4.44)$$

To enforce the  $n$  matching conditions, the matrix  $\hat{\mathbf{D}}$  is obtained from  $\mathbb{D}$  by (1) scaling to account for the shift  $[-1, 1] \rightarrow [0, T]$  by multiplying the resulting matrix by  $2/T$ , and (2) modifying the last  $n$  rows as  $[\mathbf{0}_n \ \mathbf{0}_n \ \dots \ \mathbf{I}_n]$  where  $\mathbf{0}_n$  and  $\mathbf{I}_n$  are  $n \times n$  null and identity matrices, respectively. The pattern of the product operational matrices is

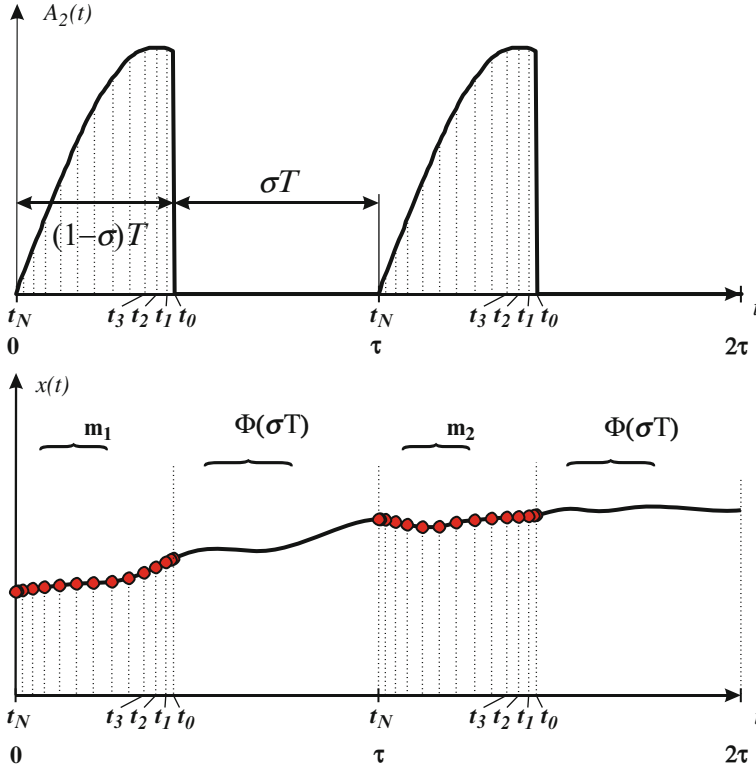
$$\hat{\mathbf{M}}_{A_1} = \begin{bmatrix} \mathbf{A}_1(t_0) & & & & \\ & \mathbf{A}_1(t_1) & & & \\ & & \ddots & & \\ & & & \mathbf{A}_1(t_{N-1}) & \\ \mathbf{0}_n & \mathbf{0}_n & \dots & \mathbf{0}_n & \mathbf{0}_n \end{bmatrix}, \quad (4.45)$$

where  $\mathbf{A}_1(t_i)$  (calculated at the  $i$ th point on the interval of length  $\tau$ ) and elements  $\mathbf{0}_n$  are  $n \times n$  matrices. Similarly,

$$\hat{\mathbf{M}}_{A_2} = \begin{bmatrix} \mathbf{A}_2(t_0) & & & & \\ & \mathbf{A}_2(t_1) & & & \\ & & \ddots & & \\ & & & \mathbf{A}_2(t_{N-1}) & \\ \mathbf{I}_n & \mathbf{0}_n & \dots & \mathbf{0}_n & \mathbf{0}_n \end{bmatrix}. \quad (4.46)$$

Here the hat (^) above the operator refers to the fact that the matrices are modified by altering the last  $n$  rows to account for the matching conditions.

In the case of nonsmooth coefficients where the matrix  $\mathbf{A}_2(t)$  vanishes for a percentage  $\sigma$  of the period  $T$ , then the DDE of (4.3) in this interval reduces to the ODE system  $\dot{\mathbf{x}} = \mathbf{A}_1(t)\mathbf{x}$  for which a transition matrix  $\Phi(t)$  may be approximated using the technique of Chebyshev polynomial expansion in [71], for example. (In certain problems such as interrupted milling, the system further reduces to one with constant coefficients, i.e.  $\dot{\mathbf{x}} = \mathbf{A}_0\mathbf{x}$ , such that the transition matrix in the subinterval where the delay vanishes is simply  $\Phi(t) = e^{\mathbf{A}_0 t}$ ). To utilize the solution  $\Phi(t)$ , we rescale the Chebyshev collocation points to account for the shift  $[-1, 1] \rightarrow [0, (1 - \sigma)T]$ , while doing the same for matrix  $\hat{\mathbf{D}}$  by multiplying  $\mathbb{D}$



**Fig. 4.6** Chebyshev collocation vectors on successive intervals for the case where  $A_2(t)$  vanishes for a percentage  $\sigma$  of the period  $T$

by  $2/((1 - \sigma)T)$ . As the matching condition between successive intervals now becomes  $\mathbf{m}_{1N} = \Phi(\sigma T)\mathbf{m}_{\phi 0}$  (Fig. 4.6), therefore, the last  $n$  rows of  $\hat{\mathbf{M}}_{A_2}$  in (4.46) are modified to  $[\Phi(\sigma T) \mathbf{0}_n \dots \mathbf{0}_n]$ .

Therefore, since  $\mathbf{U}$  is defined as the mapping of the solution at the collocation points to successive intervals as  $\mathbf{m}_1 = \mathbf{U}\mathbf{m}_\phi$  (compare with (4.4)), we obtain the approximation to the monodromy operator from (4.44) as

$$\mathbf{U} = [\hat{\mathbf{D}} - \hat{\mathbf{M}}_{A_1}]^{-1} \hat{\mathbf{M}}_{A_2}. \quad (4.47)$$

Alternatively, the inversion of  $[\hat{\mathbf{D}} - \hat{\mathbf{M}}_{A_1}]$  can be avoided by setting the determinant of  $\{\hat{\mathbf{M}}_{A_2} - \rho[\hat{\mathbf{D}} - \hat{\mathbf{M}}_{A_1}]\}$  to zero. It is seen that if  $m$  is the number of collocation points in each interval and  $n$  is the order of the original delay differential equation, then the size of the  $\mathbf{U}$  matrix (whose eigenvalues represent the approximate Floquet multipliers which are largest in absolute value) will be  $mn \times mn$ . We can achieve higher accuracy of the Floquet multipliers by increasing the value of  $m$ . A MATLAB suite of codes called DDEC which computes stability boundaries for

linear periodic DDEs with multiple discrete delays has been written and is available for download [11]. Other than for linear stability analysis of periodic DDEs, this collocation method has also been used for center manifold reduction of nonlinear periodic DDEs [15]. The Chebyshev collocation method will be illustrated in the next section through the analysis of the stability of the milling process.

## 4.5 Application to Milling Stability

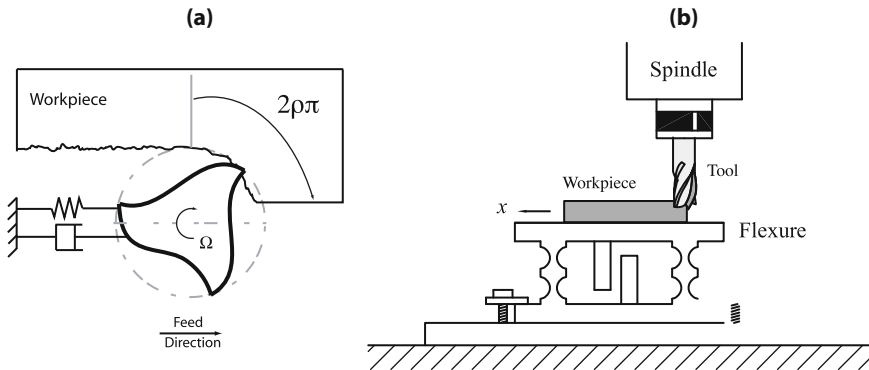
This section investigates the stability of a manufacturing process known as milling. We restrict our analysis to the case of a compliant workpiece and rigid tool (Fig. 4.7). For the sake of brevity, only the salient features of the model physical parameters are described. The interested reader is directed to references [37, 52, 55, 57, 62] for a more comprehensive model description and for the comparison of theory with experiments. The dynamics of the system under consideration can be represented as a single degree of freedom model

$$\ddot{x}(t) + 2\zeta\omega\dot{x}(t) + \omega^2x(t) = -K_s(t)b[x(t) - x(t - \tau)], \quad (4.48)$$

where  $\zeta$  is the damping ratio,  $\omega$  is the circular natural frequency, and  $b$  is the axial depth of cut. The terms of  $K_s(t)$  are

$$K_s(t) = \frac{K_t}{2m}g(t)[\sin(2\Omega t) + 0.3(1 - \cos(2\Omega t))], \quad (4.49)$$

where  $\Omega$  is the spindle rotational speed,  $g(t)$  is a unit step function (i.e.  $g_p(t) = 1$  for  $0 < \text{mod}(\Omega t) < 2\pi\rho$  and is otherwise zero), and  $\rho$  is the fraction of the spindle



**Fig. 4.7** The single degree of freedom milling system investigated in this section and in references [37, 55]. Schematic (a) is a top view and (b) is a side view

period spent cutting. The constant  $K_t$  is a cutting coefficient that scales the cutting forces in relation to the uncut chip area. The forthcoming stability results are generalized by the introduction of the following non-dimensional parameters,

$$\tilde{t} = \omega t, \quad (4.50a)$$

$$\tilde{\tau} = \omega \tau, \quad (4.50b)$$

$$\tilde{\Omega} = \frac{\Omega}{\omega}, \quad (4.50c)$$

$$\tilde{b} = \frac{bK_t}{2m\omega^2}, \quad (4.50d)$$

into (4.48). The state-space representation for the revised equation of motion is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 - \tilde{b}K_c(\tilde{t}) & -2\tilde{\zeta} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \tilde{b}K_c(\tilde{t}) & 0 \end{bmatrix} \begin{bmatrix} x_1(\tilde{t} - \tilde{\tau}) \\ x_2(\tilde{t} - \tilde{\tau}) \end{bmatrix}, \quad (4.51)$$

where the expression for  $K_c(\tilde{t})$  is

$$K_c(\tilde{t}) = g(\tilde{t}) \left( \sin 2\tilde{\Omega}\tilde{t} + 0.3 (1 - \cos 2\tilde{\Omega}\tilde{t}) \right). \quad (4.52)$$

Investigating the stability of (4.51) also requires a solution for the time interval of free vibration,  $t_f$ , or the time interval when  $g(\tilde{t}) = 0$ . For the TFEA method, the state transition matrix  $\Phi$  in (4.14) is used for this purpose, while it is also used in the Chebyshev collocation method in (4.46). Thus it maps the states of the tool as it exits out of the cut to the states of the tool as it reenters into the cut. The terms that populate the state transition matrix are

$$\Phi_{11} = \frac{\lambda_1 e^{\lambda_2 \tilde{t}_f} - \lambda_2 e^{\lambda_1 \tilde{t}_f}}{\lambda_1 - \lambda_2}, \quad (4.53)$$

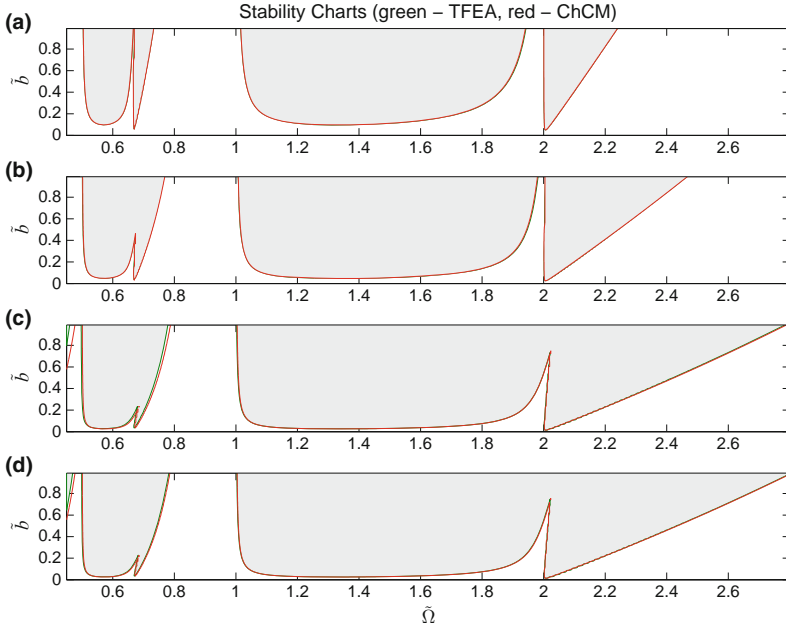
$$\Phi_{12} = \frac{e^{\lambda_1 \tilde{t}_f} - e^{\lambda_2 \tilde{t}_f}}{\lambda_1 - \lambda_2}, \quad (4.54)$$

$$\Phi_{21} = \frac{\lambda_1 \lambda_2 e^{\lambda_2 \tilde{t}_f} - \lambda_1 \lambda_2 e^{\lambda_1 \tilde{t}_f}}{\lambda_1 - \lambda_2}, \quad (4.55)$$

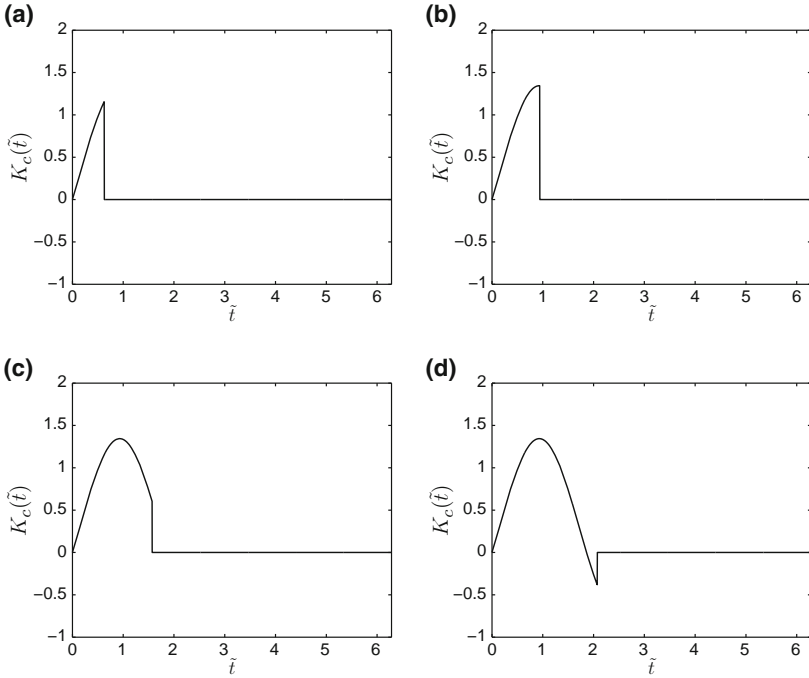
$$\Phi_{22} = \frac{\lambda_1 e^{\lambda_1 \tilde{t}_f} - \lambda_2 e^{\lambda_2 \tilde{t}_f}}{\lambda_1 - \lambda_2}, \quad (4.56)$$

where  $\tilde{t}_f = \omega t_f$  and the subscripts for each  $\Phi$ -term denote the row and column within the state transition matrix. The remaining undefined terms are  $\lambda_1 = -\zeta + \sqrt{\zeta^2 - 1}$  and  $\lambda_2 = -\zeta - \sqrt{\zeta^2 - 1}$ .

Figure 4.8 shows several stability charts computed using both the TFEA method and Chebyshev collocation method for several different values of  $\rho$ , while Fig. 4.9 shows the corresponding specific cutting forces at these same values. The stability charts were computed using three finite elements and 25 collocation points with a  $900 \times 50$  and  $300 \times 300$  grid, respectively, which results in similar computational



**Fig. 4.8** Milling process stability charts for  $\zeta = 0.003$  and (a)  $\rho = 0.10$ , (b)  $\rho = 0.15$ , (c)  $\rho = 0.25$ , and (d)  $\rho = 0.33$ . The TFEA (green) and Chebyshev collocation (red) results are almost identical. Unstable regions are shaded and stable regions are left unshaded



**Fig. 4.9** Plots of the piecewise continuous term, from (4.52), over a single period. Various cutting to noncutting times are indicated by (a)  $\rho = 0.10$ , (b)  $\rho = 0.15$ , (c)  $\rho = 0.25$ , and (d)  $\rho = 0.33$

times (around one minute on a modern laptop). It is seen that the collocation and TFEA results match very well. Further investigation reveals that the two methods are very similar in both accuracy and convergence.

## 4.6 Control of Periodic Systems with Delay using Chebyshev Polynomials

Consider the following linear time periodic delay differential system:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}_1(t)\mathbf{x}(t) + \mathbf{A}_2(t)\mathbf{x}(t - \tau) + \mathbf{B}(t)\mathbf{u}(t) \\ \mathbf{x}(t) &= \phi(t), \quad -\tau \leq t \leq 0\end{aligned}\tag{4.57}$$

which is identical to (4.3), but with the addition of a given  $n \times p$  periodic matrix  $\mathbf{B}(t)$  and a  $p \times 1$  control vector  $\mathbf{u}(t)$  (to be designed). In this section we use Chebyshev polynomial expansion for the problems of optimal (open-loop) control and delayed state feedback (closed-loop) control of 4.57. Additional details on these techniques can be found in [2, 18, 47, 50]. This work is partially based on previous work on the control of non-delay systems using Chebyshev polynomials [41, 72].

### 4.6.1 Variation of Parameters Formulation

In this section, instead of the direct method shown in Sect. 4.4.1, we use an alternative method which is based on the use of a variation of parameters formula (convolution integral). After normalizing the period to  $T = \tau = 1$ , the solution of (4.57) in the interval  $[0, 1]$  can be computed as

$$\mathbf{x}_1(t) = \Phi(t)\mathbf{x}(0) + \Phi(t) \int_0^t \Psi^T(s)[\mathbf{A}_2(s)\phi(s) + \mathbf{B}(s)\mathbf{u}(s)]ds \tag{4.58}$$

with  $\Psi^T(0) = \mathbf{I}$ . Let  $\mathbf{A}_1(t)$ ,  $\mathbf{A}_2(t)$ ,  $\mathbf{x}(t)$ ,  $\phi(t)$  be expanded in shifted Chebyshev polynomials as in Section 4.4.1. In addition, we also expand the matrices and vectors

$$\begin{aligned}\Phi(t) &= \hat{\mathbf{T}}^T(t)\mathbf{F} = \mathbf{F}'\hat{\mathbf{T}}(t), \quad \Psi(t) = \hat{\mathbf{T}}^T(t)\mathbf{P} = \mathbf{P}'\hat{\mathbf{T}}(t), \\ \mathbf{B}(t) &= \hat{\mathbf{T}}^T(t)\hat{\mathbf{B}} = \hat{\mathbf{B}}'\hat{\mathbf{T}}(t), \quad \mathbf{u}(t) = \hat{\mathbf{T}}^T(t)\mathbf{q}_1.\end{aligned}\tag{4.59}$$

Using the product and integration operational matrices defined in Section (4.4.1), Eq. (4.58) is converted into an algebraic form as

$$\mathbf{m}_1 = [\mathbf{F}\hat{\mathbf{T}}^T(1) + \hat{\mathbf{Q}}_{\mathbf{F}'}\hat{\mathbf{G}}^T\hat{\mathbf{Q}}_{\mathbf{P}'}\hat{\mathbf{Q}}_{\mathbf{A}_2}]\mathbf{m}_0 + [\hat{\mathbf{Q}}_{\mathbf{F}'}\hat{\mathbf{G}}^T\hat{\mathbf{Q}}_{\mathbf{P}'}\hat{\mathbf{Q}}_{\mathbf{B}}]\mathbf{q}_1, \tag{4.60}$$

where  $\mathbf{m}_0$  is the Chebyshev coefficient vector of  $\phi(t)$  in the delay interval  $[-1, 0]$ ,  $\mathbf{m}_1$  is the Chebyshev coefficient vector of  $\mathbf{x}(t)$  in the interval  $[0, 1]$  and  $\mathbf{q}_1$  is the Chebyshev coefficient vector of  $\mathbf{u}(t)$  in the interval  $[0, 1]$ .

Thus, the Chebyshev coefficients of the state  $\mathbf{m}_{i+1}$  are obtained on the interval  $[i, i+1]$  in terms of Chebyshev coefficients of the state  $\mathbf{m}_i$  on the interval  $[i-1, i]$  and the control input vector  $\mathbf{q}_{i+1}$  as

$$\mathbf{m}_{i+1} = \mathbf{U}\mathbf{m}_i + \mathbf{L}\mathbf{q}_{i+1} \quad (4.61)$$

where  $\mathbf{U}$  and  $\mathbf{L}$  are the matrices in (4.60). Equation (4.61) defines a recursive relationship between the Chebyshev coefficients of the state in a particular interval in terms of the Chebyshev coefficients of the state in the previous interval and the Chebyshev coefficients of the control input vector in that particular interval. The matrix  $\mathbf{U}$  is the monodromy matrix which is a finite approximation of the infinite-dimensional operator, and whose eigenvalues are identical to those of the map in (4.38). Therefore, the stability can be obtained using Chebyshev polynomial expansion of the  $\mathbf{U}$  matrix in either (4.38) or (4.60), or from Chebyshev collocation using (4.47), or from temporal finite elements using (4.14).

### 4.6.2 Finite Horizon Optimal Control via Quadratic Cost Function

We now consider the problem of finite horizon optimal control of (4.57) by minimizing the cost function

$$J = \frac{1}{2}[\mathbf{x}^T(t_f)\mathbf{S}_f\mathbf{x}(t_f) + \int_0^{t_f} [\mathbf{x}^T(t)\mathbf{Q}(t)\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}(t)\mathbf{u}(t)]dt]. \quad (4.62)$$

Here,  $t_f$  is the final time,  $\mathbf{Q}(t)$  and  $\mathbf{R}(t)$  are  $n \times n$  and  $p \times p$  symmetric positive semidefinite and symmetric positive definite periodic matrices with period  $T$ , respectively. Let  $t_f$  lie between the normalized interval of time  $[N, N+1]$ .  $\mathbf{S}_f$  is an  $n \times n$  (terminal penalty) symmetric positive semidefinite matrix. If the matrices  $\mathbf{Q}(t)$  and  $\mathbf{R}(t)$  in the  $i$ th interval are represented as  $\mathbf{Q}(t) = \hat{\mathbf{T}}^T(t)\mathbf{Q}_1$  and  $\mathbf{R}(t) = \hat{\mathbf{T}}^T(t)\mathbf{R}_1$  then utilizing product and integration operational matrices, the cost function in (4.62) can be written in the form of a quadratic function in terms of the unknown Chebyshev coefficients  $\mathbf{q}_i$  of the control input vectors over each of the defined sequence of time intervals and the known Chebyshev coefficients of the initial condition function.

The Chebyshev coefficients of optimal control vector in each of the intervals are computed by equating  $\partial J / \partial \mathbf{q}_i$  to 0,  $\forall i = 1, 2, \dots, N, N+1$  using the fact that

$$\mathbf{m}_{i+1} = \mathbf{U}^{i+1}\mathbf{m}_0 + \sum_{k=0}^i \mathbf{U}^k \mathbf{L} \mathbf{q}_{k+1}. \quad (4.63)$$

This results in a set of linear algebraic equations  $\mathbf{Z}\bar{\mathbf{q}} = \mathbf{b}$  that are solved for the unknown Chebyshev coefficients of the control vector  $\bar{\mathbf{q}} = [\mathbf{q}_1^T \mathbf{q}_2^T \dots \mathbf{q}_{N+1}^T]^T$ .

### 4.6.3 Optimal Control Using Convergence Conditions

The performance of the controlled state trajectories can be improved by imposing convergence constraints on the Chebyshev coefficients of the state vector in the instances when the objective of the control input is to suppress the overall oscillation of the state vector in finite time. In particular, a quadratic convergence condition of the form

$$\mathbf{m}_{i+1}^T \mathbf{m}_{i+1} - \mathbf{m}_i^T \mathbf{m}_i \leq -\varepsilon^2 \mathbf{m}_i^T \mathbf{m}_i \quad 0 < \varepsilon \leq 1, \quad (4.64)$$

is of great utility. Such constraints are very popular in the model predictive control methodology. It can be seen that the convergence condition imposes the  $L_2$  norm of the Chebyshev coefficients of the state to decrease in the successive intervals. Also, maximizing  $\varepsilon^2$  or minimizing  $-\varepsilon^2$  increases the speed of decay. Substituting (4.61) in (4.64) yields

$$(\mathbf{U}\mathbf{m}_i + \mathbf{L}\mathbf{q}_{i+1})^T (\mathbf{U}\mathbf{m}_i + \mathbf{L}\mathbf{q}_{i+1}) \leq (1 - \varepsilon^2) \mathbf{m}_i^T \mathbf{m}_i, \quad (4.65)$$

The Chebyshev coefficients  $\mathbf{q}_{i+1}$  of the control vector and  $\varepsilon$  are to be computed by optimizing a quadratic cost to maximize the decay rate. In particular, a non-linear optimization program is formulated and solved in each interval. We minimize a quadratic cost  $-\varepsilon^2$ , subject to the quadratic convergence condition (4.65) and a linear matching condition given by

$$\hat{\mathbf{T}}^T(0)\mathbf{m}_{i+1} = \hat{\mathbf{T}}^T(1)\mathbf{m}_i, \quad (4.66)$$

which enforces the continuity of the controlled state vector in adjacent intervals. Thus, an NLP is solved in each interval to compute the optimized values of Chebyshev coefficients of the control vector  $\mathbf{q}_{i+1}$  and  $\varepsilon$ . The Chebyshev coefficients of the state vector in each interval are computed from (4.61). It is proved in [18] that this NLP problem optimizes the state trajectories of system (4.57) to achieve zero state regulation.

Alternatively, NLP can be avoided by imposing yet another convergence condition of the form

$$\|\mathbf{m}_{i+1}\|_\infty - \|\mathbf{m}_i\|_\infty \leq \varepsilon \|\mathbf{m}_i\|_\infty, \quad 0 < \varepsilon \leq 1, \quad (4.67)$$

where  $\|\cdot\|_\infty$  represents the  $L_\infty$  norm of the argument. Therefore, an LP can be formulated and solved in every interval instead of an NLP. We minimize  $-\varepsilon$ , which in turn maximizes the decay rate of the  $L_\infty$  norm of the state vector in successive intervals subject to the convergence condition obtained by substituting (4.61) into (4.67) and a matching condition of the form of (4.66). Thus, an LP is solved in each interval to compute the Chebyshev coefficients of the control vector  $\mathbf{q}_{i+1}$  and  $\varepsilon$ . The Chebyshev coefficients of the state in each interval are then computed by using (4.61). It is proved in [18] that the LP problem optimizes the state trajectories of system (4.57) to achieve zero state regulation.

#### 4.6.4 Example: Optimal Control of a Delayed Mathieu Equation

Consider the controlled delayed Mathieu equation in state-space form

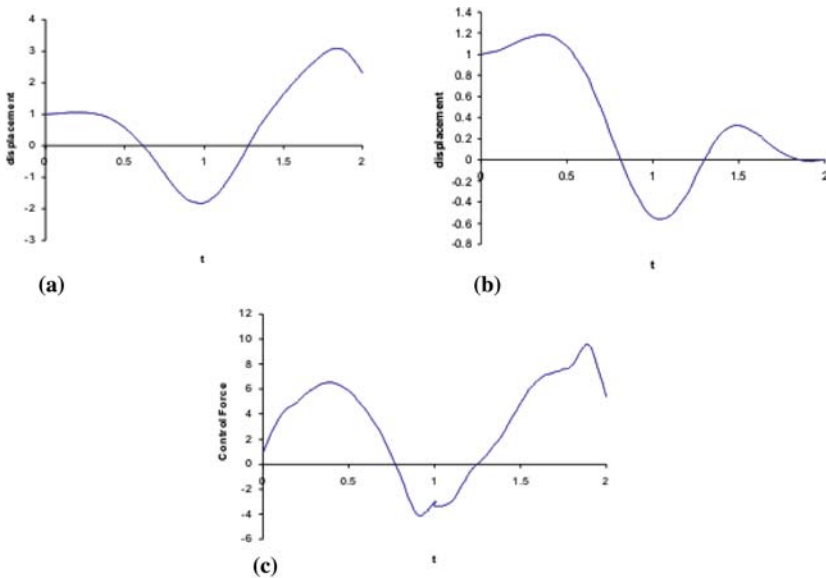
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -(a + b \cos(2\pi t)) & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -c \cos(2\pi t) & 0 \end{bmatrix} \begin{bmatrix} x_1(t-1) \\ x_2(t-1) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \quad (4.68)$$

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad -1 \leq t \leq 0,$$

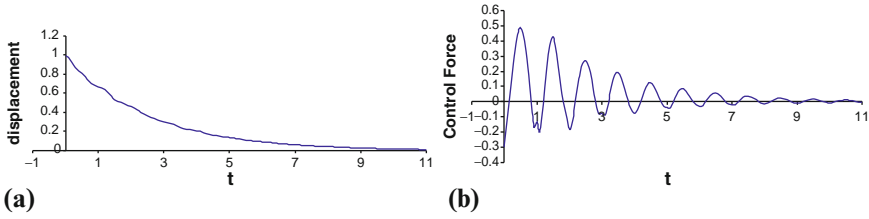
where the quadratic cost function to be minimized is

$$J = \frac{1}{2} \left\{ 10^4 \mathbf{x}^T(2) \mathbf{x}(2) + \int_0^2 [\mathbf{x}^T(t) \mathbf{x}(t) + u^2(t)] dt \right\} \quad (4.69)$$

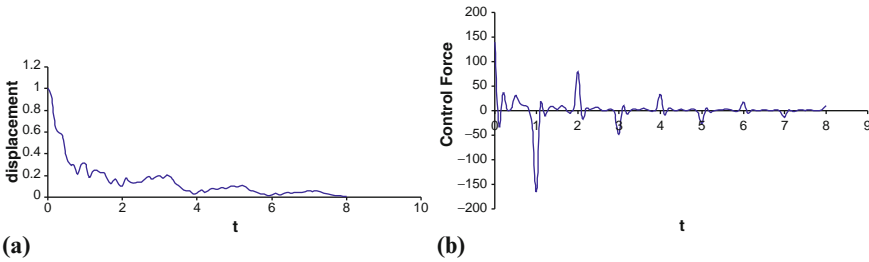
with  $\mathbf{x}(t) = [x_1(t) \ x_2(t)]^T$ . For  $a = b = 0.8\pi^2$ ,  $c = 2\pi^2$ , the uncontrolled system is unstable as the spectral radius of the monodromy matrix is larger than one. It is desired that the system state be zero at the final time  $t_f = 2.0$ . Thus, there are two distinct intervals  $[0, 1]$  and  $[1, 2]$ , to be considered in the optimization procedure which is carried out only once for the entire length of time. The uncontrolled and controlled system response (displacement  $x_1(t)$ ) and control effort obtained on the two intervals using 8 Chebyshev polynomials are as shown in Fig. 4.10.



**Fig. 4.10** (a) Uncontrolled displacement for (4.68), (b) Controlled displacement and (c) control force  $u(t)$  for (4.68) using quadratic cost function. Reprinted from *Optimal Control Applications and Methods*, vol. 27, pp. 123–136, “Optimal Control of Parametrically Excited Linear Delay Differential Systems via Chebyshev Polynomials,” by V. Deshmukh, *et al.*, 2006. ©John Wiley & Sons Limited. Reproduced with permission



**Fig. 4.11** (a) Controlled displacement and (b) control force  $u(t)$  for (4.68) using NLP. Reprinted from *Optimal Control Applications and Methods*, vol. 27, pp. 123-136, “Optimal Control of Parametrically Excited Linear Delay Differential Systems via Chebyshev Polynomials,” by V. Deshmukh, et al., 2006. ©John Wiley & Sons Limited. Reproduced with permission



**Fig. 4.12** (a) Controlled displacement and (b) control force  $u(t)$  for (4.68) using LP. Reprinted from *Optimal Control Applications and Methods*, vol. 27, pp. 123-136, “Optimal Control of Parametrically Excited Linear Delay Differential Systems via Chebyshev Polynomials,” by V. Deshmukh, et al., 2006. ©John Wiley & Sons Limited. Reproduced with permission

The NLP is now solved for (4.68) and optimized values of  $\mathbf{q}_{i+1}$  and  $\varepsilon$  are computed until the displacement  $x_1(t)$  approaches zero for the entire interval. The controlled response (displacement  $x_1(t)$ ) and control effort are as shown in Fig. 4.11. Also, the LP is solved for (4.68) and the optimized values of  $\mathbf{q}_{i+1}$  and  $\varepsilon$  are computed until the displacement  $x_1(t)$  approaches zero for the entire interval. The controlled state response (displacement  $x_1(t)$ ) and control effort are as shown in Fig. 4.12. It can be seen that the control vector obtained using the LP formulation renders faster decaying controlled trajectories than does the one obtained using the NLP formulation. Both the NLP and LP formulations implicitly minimize a cost function including the state vector.

#### 4.6.5 Delayed State Feedback Control

The next problem we address is the symbolic delayed feedback control of the system given in (4.57). Assuming the uncontrolled system is unstable, it is desired to obtain a delayed state feedback control given by

$$\mathbf{u}(t) = \mathbf{K}(t, \hat{\mathbf{k}})\mathbf{x}(t - \tau), \quad (4.70)$$

where  $\mathbf{K}(t, \hat{\mathbf{k}}) = \mathbf{K}(t + T, \hat{\mathbf{k}})$  is a  $k \times n$  periodic gain matrix to be found symbolically in terms of a vector  $\hat{\mathbf{k}}$  of control gains which asymptotically stabilize (4.57). It is assumed that the present state  $\mathbf{x}(t)$  is not available for feedback. Such a situation arises in practical control problems involving the delay in sensing, measurement or reconstruction of the state. Expanding  $\mathbf{B}(t)\mathbf{K}(t, \hat{\mathbf{k}}) = \hat{\mathbf{T}}^T(t)\mathbf{V}(\hat{\mathbf{k}})$  in terms of shifted Chebyshev polynomials valid on the interval  $[0, 1]$  and inserting into (4.58) yields the solution in the  $i$ th interval as

$$\mathbf{x}_i(t) = \hat{\mathbf{T}}^T(t)[\mathbf{F}\hat{\mathbf{T}}^T(1)\mathbf{m}_{i-1} + \hat{\mathbf{Q}}_{\mathbf{F}'}\hat{\mathbf{G}}^T\hat{\mathbf{Q}}_{\mathbf{P}'}(\hat{\mathbf{Q}}_{\mathbf{A}_2} + \hat{\mathbf{Q}}_{\mathbf{V}}(\hat{\mathbf{k}}))\mathbf{m}_{i-1}] = \hat{\mathbf{T}}^T(t)\mathbf{m}_i \quad (4.71)$$

$$0 \leq \tilde{t} \leq 1, \quad i-1 \leq t \leq i,$$

where  $\mathbf{m}_i$  is the Chebyshev coefficient vector in the interval  $[i-1, i]$  and  $\mathbf{x}_i(i-1) = \mathbf{x}_{i-1}(i)$ . Defining the closed-loop monodromy matrix as

$$\mathbf{U}(\hat{\mathbf{k}}) = \mathbf{F}\hat{\mathbf{T}}^T(1) + \hat{\mathbf{Q}}_{\mathbf{F}'}\hat{\mathbf{G}}^T\hat{\mathbf{Q}}_{\mathbf{P}'}(\hat{\mathbf{Q}}_{\mathbf{A}_2} + \hat{\mathbf{Q}}_{\mathbf{V}}(\hat{\mathbf{k}})), \quad (4.72)$$

(4.71) reduces to

$$\mathbf{m}_i = \mathbf{U}(\hat{\mathbf{k}})\mathbf{m}_{i-1}. \quad (4.73)$$

Matrix  $\mathbf{U}(\hat{\mathbf{k}})$  in (4.73) is the monodromy matrix of the closed-loop controlled system and is dependent on the vector  $\hat{\mathbf{k}}$  of control gains. It advances the solution forward by one period and its spectral radius should be within the unit circle [13] for asymptotic stability of (4.57). The problem is to find the control input in the form of a delayed state feedback as  $\mathbf{u}(t) = \mathbf{K}(t, \hat{\mathbf{k}})\mathbf{x}(t - \tau)$ , so that the closed loop monodromy matrix  $\mathbf{U}(\hat{\mathbf{k}})$  has spectral radius within the unit circle. The advantage of using delayed state feedback over present state feedback is that the symbolic monodromy matrix of the closed loop system is linear with respect to the controller parameters. Such a property would not be obtained if one uses present state feedback.

The above procedure of forming matrix  $\mathbf{U}(\hat{\mathbf{k}})$  can be performed in symbolic software such as *Mathematica*. To compute the unknown parameters  $\hat{\mathbf{k}}$ , one has to convert the characteristic polynomial of discrete time map (4.72) to the characteristic polynomial of the equivalent continuous time system. By applying the Routh-Hurwitz criterion to that characteristic polynomial, a Routh-Hurwitz matrix is constructed. For asymptotic stability, all the determinants of leading minors of this Routh-Hurwitz matrix have to be positive. Imposing this condition on each of the determinants, one obtains nonlinear inequalities describing regions in the parameter space. The common region of intersection between these individual regions is the desired stability region in the parameter space. To choose the values of the controller parameters, an appropriate point in the stable region of the parameter space must be selected.

### 4.6.6 Example: Delayed State Feedback Control of the Delayed Mathieu Equation

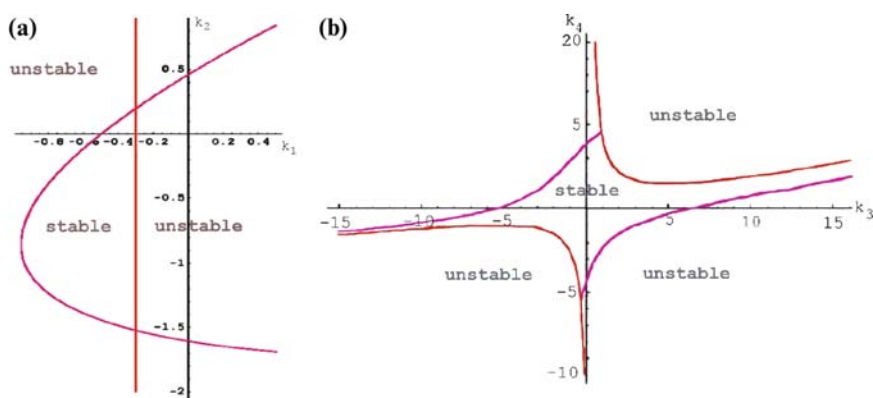
Reconsider the delayed Mathieu equation in (4.68) where  $a = 0.2$ ,  $b = 0.1$ , and  $c = -0.5$ . Following the procedure described in Sect. 4.4.1, the monodromy matrix  $\mathbf{U}$  of the uncontrolled version of (4.68) is computed, which is found to be unstable, as its spectral radius is larger than unity. In order to obtain the asymptotic stability of the controlled system, we design a delayed state feedback controller with a constant gain matrix given as

$$\mathbf{K}(\hat{\mathbf{k}}) = [k_1 \ k_2]. \quad (4.74)$$

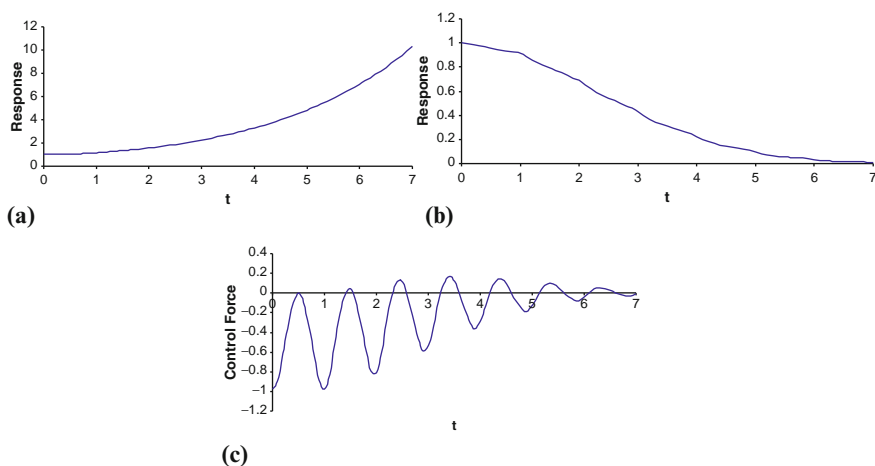
The close loop monodromy matrix  $\mathbf{U}(k_1, k_2)$  is formed symbolically by using the methodology described above. By using  $m = 4$  shifted Chebyshev polynomials and applying the continuous time stability criterion (Routh-Hurwitz) after transforming the characteristic equation from unit circle stability to left-half plane stability, we obtain the stability boundary. The stability region in the parameter space  $[k_1, k_2]$  is shown in Fig. 4.13a. There are  $nm = 8$  determinants of the  $nm$  leading minors of the symbolic Routh-Hurwitz matrix. The stability boundary is determined by all the  $nm$  curves in the parameter space given by the  $nm$  determinants. In Fig. 4.13a, however, we plot only the region common to  $nm$  curves, which is the desired region of stability in the parameter space. We select the parameters  $k_1 = -0.49$ ,  $k_2 = -0.5$  from the stable region and apply the controller to the system. The asymptotic stability of the dynamic system is clearly seen from the spectral radius of closed loop monodromy matrix given by 0.604125.

We can enlarge the stability region of the system by designing a time-periodic gain matrix as

$$\mathbf{K}(t, \hat{\mathbf{k}}) = [-0.49 + k_3 \cos(2\pi t) \quad -0.5 + k_4 \sin(2\pi t)], \quad (4.75)$$



**Fig. 4.13** The stability region for (4.68) with (a) constant gain matrix and (b) periodic gain matrix with  $k_1 = -0.49$ ,  $k_2 = -0.5$ . Reprinted from *Communications in Nonlinear Science and Numerical Simulation*, vol. 10, pp. 479–497, “Delayed State Feedback and Chaos Control for Time-Periodic Systems via a Symbolic Approach,” by H. Ma et al. ©2005, with permission from Elsevier



**Fig. 4.14** (a) The uncontrolled response for the (4.68), (b) Controlled response and (c) control force  $u(t)$  of (4.68) using delayed state feedback control with a time-periodic gain matrix. Reprinted from *Communications in Nonlinear Science and Numerical Simulation*, vol. 10, pp. 479–497, “Delayed State Feedback and *Chaos* Control for Time-Periodic Systems via a Symbolic Approach,” by H. Ma et al. ©2005, with permission from Elsevier

where the average parts are the values of  $k_1$  and  $k_2$  selected above. Applying the same procedure, we obtain the stability region as shown in Fig. 4.13b. Again, we plot only the intersection of  $nm = 8$  curves, which are the determinants of the symbolic Routh-Hurwitz matrix obtained for this case. The spectral radius of closed loop monodromy matrix is improved to 0.5786 with parameters  $k_3 = -0.49$  and  $k_4 = -0.5$  and the stability region is clearly enlarged. The uncontrolled and controlled responses and the control effort of Eq. (4.68) obtained by applying the periodic gain matrix in (4.75) are illustrated in Fig. 4.14.

## 4.7 Discussion of Chebyshev and TFEA Approaches

In this fourth chapter, the authors have presented two different approaches, based on Chebyshev polynomials and temporal finite elements, respectively, for investigating the stability behavior of linear time-periodic DDEs. Both approaches are applicable to a broader class of systems that may be written in the form of a state space model. Two different Chebyshev-based methods, that of polynomial expansion and collocation, were explained. Both Chebyshev and TFEA methods were used to produce stability charts for delayed Mathieu equations as well as a model for the milling process. Chebyshev polynomial expansion was also used for the optimal and delayed state feedback control of periodic systems with delay. The discretization methods presented in this chapter complement the semi-discretization method discussed in the fifth chapter.

Specifically, compared with the traditional numerical simulation method for stability analysis of delayed systems, the current techniques are much simpler as all of the computation is implemented by matrix manipulation. And since an approximation to the monodromy operator is found, they are also useful for designing controllers to stabilize the original system. Also, much of the information needed to set up the problem can be stored in the computer in advance. The “product” and “integration” matrices associated with the shifted Chebyshev polynomials can be readily constructed from the general expressions. In general, the periodic terms in  $\mathbf{A}(t)$  have the forms  $\sin(n\pi t/T)$  and/or  $\cos(n\pi t/T)$ . The expansions of these quantities can be made a part of the subroutine, and one does not have to compute the expansion coefficients each time. The entire computation process can be automated rather easily.

Two different formulas for the computation of the approximate  $\mathbf{U}$ , whose size is determined by the number of polynomials employed, were given. The first one used the direct integral form of the original system in state space form while the other one used a convolution integral (variation of parameters) formulation. An error analysis was presented, which allows the number of polynomials employed in the approximation to be selected in advance for a desired tolerance. An extension of the Chebyshev-based methods to the case where the delay and parametric periods are commensurate was not shown here but its implementation is straightforward as in the DDEC collocation code [11].

When one compares the two Chebyshev-based methods, however, (i.e., polynomial expansion and collocation), one can quickly see that the collocation method is more efficient. This can be seen by comparing the forms for the monodromy matrix  $\mathbf{U}$  in (4.38), (4.47), (4.60) while observing the differences in the sparseness of the corresponding operational matrices. In [12] it is shown that the eigenvalues of the  $\mathbf{U}$  matrix obtained via collocation are spectrally convergent to the exact multipliers of the monodromy operator. Also, in that paper it is shown that new eigenvalue perturbation and *a posteriori* estimation techniques give computable error bounds on the eigenvalue approximation error. One would obviously like an *a priori* proof of the spectral convergence of the collocation method when the coefficients of the DDE are analytic, but this is open. That is, one would want to show spectral convergence, as the degree of polynomial collocation increases, for the approximate solutions of DDEs and for the approximate eigenvalues of the monodromy operator.

An important consideration for the TFEA approach is the choice of trial and weighting functions. In particular, the illustration of Fig. 4.1 highlights the fact that the trial functions were chosen so that a single coefficient of the assumed solution would equal the state variable at the beginning, middle, and end of the temporal element. This provides three constraints for interpolating the polynomials with the remaining constraints being used to ensure orthogonality and meet user-preferences. We have found (4.7a–4.7c) to be a particularly useful set of polynomials because higher order polynomials sometimes caused round-off errors for smaller temporal elements (i.e. seventh order polynomials would require that  $t_j^7$  remain within the machine precision). Our choice of Legendre polynomials for weighting functions was based upon a few factors. First, we made comparisons with the weighting

functions used in recent works [6] and also implemented a Galerkin procedure. In both cases, we found that converged stability boundaries were obtained with fewer elements when using Legendre polynomials. However, we make no claim that the current set of trial and weighting functions are the most optimum set of polynomials. Another consideration is the number of weighting functions to apply. While our preference was to use a combination of two weighting functions and three trial functions, which will always keep the  $\mathbf{R}$  and  $\mathbf{H}$  matrices square, it is certainly possible to use more weighting functions.

Since the TFEA method is a discretization approach, a brief discussion of solution convergence seems necessary. Following the work in spatial finite elements, we recognize three different approaches for solution convergence: (1) the number of temporal elements can be increased (h-convergence); (2) the polynomial order may be increased (p-convergence); or (3) both the polynomial order and number of elements can be increased (hp-convergence). While a much more comprehensive discussion is offered in reference [24], let it suffice to say that tracking the characteristic multipliers while increasing the number of elements or polynomial order may be used for convergence. Here, the authors simply increased the number of temporal elements for each figure until a converged result was obtained.

Computational efficiency and implementation are also issues worthy of some discussion. With regards to implementation, the integrals of (4.11a), (4.11b), (4.15a), (4.15b) were all performed symbolically with the software package MATLAB. These terms were then inserted into stability algorithms that varied the control parameters while recording the largest characteristic multiplier for the formed matrices. With regards to computational efficiency, the algorithm time to produce stability results for a  $900 \times 50$  (TFEA) and  $300 \times 300$  grid (Chebyshev collocation) of control parameters was typically around one minute for a modern laptop computer using either Chebyshev collocation or TFEA approaches.

In summary, the presented Chebyshev and TFEA methods provide computationally efficient ways to query the asymptotic stability of periodic DDEs written in the form of a state space model. We have some questions for further research about the monodromy operator  $U$  itself. For example, under what conditions does  $U$  actually diagonalize? Also, what *a priori* estimates can be computed for the conditioning of its eigenvalue problem? What can generally be said about the pseudospectra of  $U$ ? Also, are the eigenfunctions of  $U$  relatively normal or can non-normality lead to instability even if all multipliers have less than unity modulus? These and other questions are for future research.

**Acknowledgments** As one might expect, the results of this chapter required the support, assistance, and efforts of many individuals. Thus we would like to recognize those who have influenced and shaped our work in this area. First, we thank Gábor Stépán, who has put forth many interesting questions, influenced, and given inspiration to the work of both authors. Next, we wish to extend our deepest gratitude to our Ph.D. advisors Professors S. C. Sinha and P. V. Bayly, who are great colleagues, have become wonderful friends, and continue to share their intellectual insights. Lastly, both authors would like to thank their colleagues and graduate students. Dr. Butcher recognizes the contributions of colleagues Ed Bueler, Venkatesh Deshmukh, and Zsolt Szabó, the helpful advice of David Gilsinn and Gábor Stépán, as well as the results of students Victoria Averina, Haitao Ma,

Praveen Nindujarla and Oleg Bobrenkov that were invaluable in this research. In addition, Dr. Mann would like to thank colleagues Tamás Insperger, Gábor Stépán, Tony Schmitz, Keith Young, Amy Helvey, and Ryan Hanks as well as former students Nitin Garg, Mike Koplow, Ryan Carter, Jian Liu, Bhavin Patel, Ben Edes, and Firas Khasawneh for their contributions.

## References

1. Argyris, J. H., and Scharpf, D. W., 'Finite Elements in Time and Space', *Aeronaut. J. Roy. Soc.* **73**, 1969, 1041–1044.
2. Averina, V., *Symbolic stability of delay differential equations*, MS Thesis, Dept. of Mathematical Sciences, University of Alaska Fairbanks, 2002.
3. Balachandran, B., 'Non-Linear Dynamics of Milling Process', *Proceedings of the Royal Society of London A* **359**, 2001, 793–819.
4. Balachandran, B. and Gilsinn, D., 'Nonlinear Oscillations of Milling', *Mathematical and Computer Modelling of Dynamical Systems* **11**, 2005, 273–290.
5. Batzel, J. J., and Tran, H. T., 'Stability of the Human Respiratory Control System. Part I: Analysis of a Two-Dimensional Delay State-Space Model', *J. Mathemat. Biol.* **41**, 2000, 45–79.
6. Bayly P. V., Halley, J. E., Mann, B. P., and Davis, M. A., 'Stability of Interrupted Cutting by Temporal Finite Element Analysis', *J. Manufact. Sci. Eng.* **125**, 2003, 220–225.
7. Bellen, A., 'One-step collocation for delayed differential equations', *J. Comp. Appl. Math.* **10**, 1984, 275–283.
8. Bellman, R., and Cooke, K., *Differential-Difference Equations*, Academic, 1963.
9. Borri, M., Bottasso, C., and Mantegazza, P., 'Basic Features of the Time Finite Element Approach for Dynamics', *Meccanica* **27**, 1992, 119–130.
10. Budak, E., and Altintas, Y., 'Analytical Prediction of Chatter Stability in Milling – Parts I and II', *J. Dynam. Sys. Measure. Cont.* **120**, 1998, 22–36.
11. Bueler, E., 'Guide to DDEC: Stability of linear, periodic DDEs using the DDEC suite of Matlab codes', <http://www.dms.uaf.edu/bueler/DDEcharts.htm>, 2005.
12. Bueler, E., 'Error bounds for approximate eigenvalues of periodic-coefficient linear delay differential equations', *SIAM J. Num. Anal.* **45**, 2007, 2510–2536.
13. Butcher, E. A., Ma, H., Bueler, E., Averina, V., and Szabó, Z., 'Stability of Linear Time-Periodic Delay-Differential Equations via Chebyshev Polynomials', *International J. Num. Meth. Eng.* **59**, 2004, 895–922.
14. Butcher, E. A., Nindujarla, P., and Bueler, E., 'Stability of Up- and Down-Milling Using Chebyshev Collocation Method', *proceedings of 5th International Conference on Multibody Systems, Nonlin. Dynam. Cont.* ASME DETC05, Long Beach, CA, Sept. 24–28, 2005.
15. Butcher, E. A., Deshmukh, V., and Bueler, E., 'Center Manifold Reduction of Periodic Delay Differential Systems', *proceedings of 6th International Conference on Multibody Systems, Nonlinear Dynamics, and Control*, ASME DETC07, Las Vegas, NV, Sept. 4–7, 2007.
16. Butcher, E. A., Bobrenkov, O. A., Bueler, E., and Nindujarla, P., 'Analysis of Milling Stability by the Chebyshev Collocation Method: Algorithm and Optimal Stable Immersion Levels', *J. Comput. Nonlin. Dynam.*, in press.
17. Clenshaw, C. W., 'The Numerical Solution of Linear Differential Equations in Chebyshev Series', *Proc. Cambridge Philosophical Society* **53**, 1957, 134–149.
18. Deshmukh, V., Ma, H., and Butcher, E. A., 'Optimal Control of Parametrically Excited Linear Delay Differential Systems via Chebyshev Polynomials', *Opt. Cont. Appl. Meth.* **27**, 2006, 123–136.
19. Deshmukh, V., Butcher, E. A., and Bueler, E., 'Dimensional Reduction of Nonlinear Delay Differential Equations with Periodic Coefficients using Chebyshev Spectral Collocation', *Nonlin. Dynam.* **52**, 2008, 137–149.
20. Elliot, D., 'A Chebyshev Series Method for the Numerical Solution of Fredholm Integral Equations', *Comp. J.* **6**, 1963, 102–111.

21. Engelborghs, K., Luzyanina, T., in T Hout, K. J., and Roose, D., 'Collocation Methods for the Computation of Periodic Solutions of Delay Differential Equations', *SIAM J. Sci. Comput.* **22**, 2000, 1593–1609.
22. Fox, L., and Parker, I.B., *Chebyshev Polynomials in Numerical Analysis*, Oxford Univ. Press, London, 1968.
23. Fried, I., 'Finite Element Analysis of Time Dependent Phenomena', *Am. Inst. Aeron. Astron. J.* **7**, 2002, 1170–1173.
24. Garg, N. K., Mann, B. P., Kim, N. H., Kurdi, M. H., 'Stability of a Time-Delayed System With Parametric Excitation', *J. Dynam. Syst. Measur. Cont.* **129**, 2007, 125–135.
25. Gilsinn, D. E., and Potra, F. A., 'Integral Operators and Delay Differential Equations', *J. Int. Equ. Appl.* **18**, 2006, 297–336.
26. Gouskov, A. M., Voronov, S. A., Butcher, E. A., and Sinha, S. C., 'Nonconservative Oscillations of a Tool for Deep Hole Honing', *Comm. Nonlin. Sci. Num. Simulation* **11**, 2006, 685–708.
27. Hahn, W., 'On difference differential equations with periodic coefficients', *J. Mathemat. Anal. Appl.* **3**, 1961, 70–101.
28. Halanay, A., *Differential Equations: Stability, Oscillations, Time Lags*, Academic Press, New York, 1966.
29. Hale, J. K., and Verduyn Lunel, S. M., *Introduction to Functional Differential Equation*, Springer, New York, 1993.
30. Hayes, N. D., 'Roots of the Transcendental Equations Associated with Certain Differential-Difference Equations', *J. Lond. Mathemat. Soci.* **25**, 1950, 226–232.
31. Horng, I.-R., and Chou, J.-H., 'Analysis, Parameter Estimation and Optimal Control of Time-Delay Systems via Chebyshev Series', *Int. J. Cont.* **41**, 1985, 1221–1234.
32. Hsu, C. S., and Bhatt, S. J., 'Stability Charts for Second-Order Dynamical Systems with Time Lag', *J. Appl. Mech.* **33E**, 1966, 119–124.
33. Insperger, T., and Stépán, G., 'Remote Control of Periodic Robot Motion', *Proceedings of 13th CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators*, 2000, 197–203.
34. Insperger, T., and Stépán, G., 'Comparison of the Stability Lobes for Up- and Down-Milling', *Proceedings of Dynamics and Control of Mechanical Processing Workshop*, 2nd Workshop, 2001, 53–57, Budapest University of Technology and Economics, Budapest.
35. Insperger, T., and Stépán, G., 'Semi-Discretization Method for Delayed Systems', *Int. J. Num. Meth. Engr.* **55**, 2002, 503–518.
36. Insperger, T., and Stépán, G., 'Stability chart for the delayed Mathieu equation', *Proc. R. Soc., Math. Physic. Eng. Sci.* **458**, 2002, 1989–1998.
37. Insperger T., Mann, B. P., Stépán, G., and Bayly, P. V., 'Stability of Up-Milling and Down-Milling, Part 1: Alternative Analytical Methods', *Int. J. Machine Tools Manufacture* **43**, 2003, 25–34.
38. Insperger T., and Stépán, G., 'Updated Semi-Discretization Method for Periodic Delay-Differential Equations with Discrete Delay', *Int. J. Num. Meth. Eng.* **61**, 2004, 117–141.
39. Insperger, T., and Stépán, G., 'Stability Analysis of Turning with Periodic Spindle Speed Modulation via Semi-Discretization', *J. Vibr. Cont.* **10**, 2004, 1835–1855.
40. Ito, K., Tran, H. T., and Manitius, A., 'A Fully-Discrete Spectral Method for Delay-Differential Equations', *SIAM J. Numer. Anal.* **28**, 1991, 1121–1140.
41. Jaddu, H., and Shimenura, E., 'Computation of Optimal Control Trajectories Using Chebyshev Polynomials: Parameterization and Quadratic Programming', *Opt. Cont. Appl. Meth.* **20**, 1999, 24–42.
42. Jemielniak, K., and Widota, A., 'Suppression of Self-Excited Vibration by the Spindle Speed Variation Method', *Int. J. Mach. Tool Desi. Res.*, **24**, 1984, 207–214.
43. Kálmar-Nagy, T., 'A New Look at the Stability Analysis of Delay Differential Equations', *Proceedings of International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Long Beach, California, DETC2005-84740, ASME, 2005.

44. Long, X.-H., and Balachandran, B., 'Milling dynamics with a variable time delay'. *Proceedings of ASME, IMECE 2004*, Anaheim, CA, November 13 - 19, Paper No. IMECE 2004-59207.
45. Long, X.-H., and Balachandran, B., 'Stability analysis of a variable spindle speed milling process', *proceedings of 5th International Conference on Multibody Systems, Nonlin. Dynam. Contr.*, ASME DETC05, Long Beach, CA, Sep. 24-28, 2005.
46. Long, X.-H., and Balachandran, B., 'Stability Analysis for Milling Process', *Nonlin. Dynam.*, **49**, 2007, 349–359.
47. Ma, H., *Analysis and Control of Time-periodic Systems with Time Delay via Chebyshev Polynomials*, MS Thesis, Dept. of Mechanical Engineering, University of Alaska Fairbanks, 2003.
48. Ma, H., Butcher, E. A., and Bueler, E., 'Chebyshev Expansion of Linear and Piecewise Linear Dynamic Systems with Time Delay and Periodic Coefficients Under Control Excitations', *J. Dynam. Syst. Measurement Cont.* **125**, 2003, 236–243.
49. Ma, H., and Butcher, E. A., 'Stability of Elastic Columns Subjected to Periodic Retarded Follower Forces', *J. Sound Vibration* **286**, 2005, 849–867.
50. Ma, H., Deshmukh, V., Butcher, E. A., and Averina, V., 'Delayed State Feedback and Chaos Control for Time-Periodic Systems via a Symbolic Approach', *Comm. Nonlin. Sci. Num. Simula.* **10**, 2005, 479–497.
51. Mann, B. P., Insperger, T., Bayly, P. V., and Stépán, G., 'Stability of Up-Milling and Down-Milling, Part 2: Experimental Verification', *International J. of Machine Tools and Manufacture* **43**, 2003, 35–40.
52. Mann, B. P., Bayly, P. V., Davies, M. A., and Halley, J. E., 'Limit Cycles, Bifurcations, and Accuracy of the Milling Process', *J. Sound Vibration* **277**, 2004, 31–48.
53. Mann, B. P., Garg, N. K., Young, K. A., and Helvey, A. M., 'Milling bifurcations from structural asymmetry and nonlinear regeneration', *Nonlin. Dynam.* **42**, 2005, 319–337.
54. Mann, B. P., Young, K. A., Schmitz, T. L., and Dilley, D. N., 'Simultaneous stability and surface location error predictions in milling', *J. Manufact. Sci. Eng.* **127**, 2005, 446–453.
55. Mann, B. P., and Young, K. A., 'An empirical approach for delayed oscillator stability and parametric identification', *Proc. R. Soc. A* **462**, 2006, 2145–2160.
56. Mann, B. P., and Patel, B., 'Stability of delay equations written as state space models', *J. Vibra. Cont.*, invited paper for a special issue on delay systems, accepted in 2007.
57. Mann, B. P., Edes, B. T., Young, Easley, S. J., K. A., and Ma, K., 'Surface location error and chatter prediction for helical end mills,' *Int. J. Machine Tools Manufacture*, **48**, 2008, 350–361.
58. Mathieu, E., *Memoire sur le Mouvement Vibratoire d'une Membrane de Forme Elliptique*, *J. Math* **13**, 1868, 137–203.
59. Nayfeh, A. H., and Mook, D. T., *Nonlinear Oscillations*, Wiley, New York, 1979.
60. Olgac, N., Sipahi, R., 'A Unique Methodology for Chatter Stability Mapping in Simultaneous Machining', *J. Manufact. Sci. Eng.* **127**, 2005, 91–800.
61. Olgac, N., and Sipahi, R., 'Dynamics and Stability of Variable-Pitch Milling', *J. Vibrat. Cont.* **13**, 2007, 1031–1043.
62. Patel, B., Mann, B. P., and Young, K. A., 'Uncharted islands of chatter instability in milling', *Int. J. of Machine Tools Manufact.* **48**, 2008, 124–134.
63. Peters, D. A., and Izadpanah, 'Hp-Version Finite Elements for the Space-Time Domain', *Comput. Mech.* **3**, 1988, 73–78.
64. Pontryagin, L. S., 'On the zeros of some elementary transcendental functions', *Izv. Akad. Nauk SSSR* **6**, 1942, 115–134.
65. Reddy, J. N., *An Introduction To The Finite Element Method*, McGraw-Hill, Inc., New York, NY, 2nd edition, 1993.
66. Schmitz, T. L., Mann, B. P., 'Closed Form Solutions for the Prediction of Surface Location Error in Milling', *Int. Journal Mach. Tools Manufact.* **46**, 2006, 1369–1377.
67. Segalman, D. J., and Redmond, J., 'Chatter suppression through variable impedance and smart fluids', *Proc. SPIE Symposium on Smart Structures and Materials, SPIE* **2721**, 1996, 353–363.
68. Segalman, D. J., and Butcher, E. A., 'Suppression of Regenerative Chatter via Impedance Modulation', *J. Vibra. Cont.* **6**, 2000, 243–256.

69. Sexton, J. S., and Stone, B. J., 'The stability of machining with continuously varying spindle speed', *Annals of the CIRP* **27**, 1978, 321–326.
70. Champine, L. F., and Thompson, S., 'Solving DDEs in MATLAB', *Applied Numerical Mathematics* **37**, 2001, 441–458.
71. Sinha, S. C., and Wu, D.-H., 'An efficient computational scheme for the analysis of periodic systems', *J. Sound Vibration* **151**, 1991, 91–117.
72. Sinha, S. C., Gowrdon, E., and Zhang, Y., 'Control of Time-Periodic Systems via Symbolic Computation with Application to Chaos Control', *Communications in Nonlinear Science and Numerical Simulation* **10**, 2005, 835–854.
73. Snyder, M. A., *Chebyshev Methods in Numerical Approximation*, Prentice-Hall, Englewood Cliffs, N. J., 1986.
74. Stépán, G., *Retarded Dynamical Systems: Stability and Characteristic Functions*, Longman Scientific and Technical, Harlow, UK, 1989.
75. Stépán, G., 'Vibrations of Machines Subjected to Digital Force Control', *International J. Solids and Structures* **38**, 2001, 2149–2159.
76. Stépán, G., 'Modelling Nonlinear Regenerative Effects in Metal Cutting', *Philosophical Transactions of the Royal Society of London A* **359**, 2001, 739–757.
77. Stokes, A., 'A Floquet Theory for Functional Differential Equations', *Proc. Nat. Acad. Sci. U.S.A.* **48**, 1962, 1330–1334.
78. Szalai, R., and Stépán, G., 'Lobes and Lenses in the Stability Chart of Interrupted Turning', *J. Computational and Nonlinear Dynamics* **1**, 2006, 205–211.
79. Szalai, R., Stépán, G., and Hogan, S. J., 'Continuation of Bifurcations in Periodic Delay-Differential Equations Using Characteristic Matrices', *SIAM J. Sci. Comp.* **28**, 2006, 1301–1317.
80. Trefethen, L. N., *Spectral Methods in Matlab*, SIAM Press, Philadelphia, 2000.
81. Van der Pol, F., and Strutt, M. J. O., 'On the Stability of the Solutions of Mathieu's Equation', *Philos. Mag. J. Sci.*, **5**, 1928, 18–38.
82. Wright, K., 'Chebyshev collocation methods for ordinary differential equations', *Computer Journal* **6**, 1964, 358–363.
83. Zhang, J., and Sun, J.-Q., 'Robustness analysis of optimally designed feedback control of linear periodic systems with time-delay', *proceedings of 6th International Conference on Multi-body Systems, Nonlinear Dynamics, and Control*, ASME DETC07, Las Vegas, NV, Sept. 4–7, 2007.
84. Zhao, M.-X., and Balachandran, B., 'Dynamics and Stability of the Milling Process', *Int. J. Solids Struct.* **38**, 2001, 2233–2248.

# Chapter 5

## Systems with Periodic Coefficients and Periodically Varying Delays: Semidiscretization-Based Stability Analysis

Xinhua Long, Tamás Insperger, and Balakumar Balachandran

**Abstract** In this chapter, delay differential equations with constant and time-periodic coefficients are considered. The time delays are either constant or periodically varying. The stability of periodic solutions of these systems are analyzed by using the semidiscretization method. By employing this method, the periodic coefficients and the delay terms are approximated as constants over a time interval, and the delay differential system is reduced to a set of linear differential equations in this time interval. This process helps to define a Floquet transition matrix that is an approximation to the infinite-dimensional monodromy operator. Information on the stability of periodic solutions of the delay differential system is obtained from analysis of the eigenvalues of the finite linear operator. As illustrative examples, stability charts are constructed for systems with constant delays as well as time-varying delays. The results indicate that a semidiscretization-method-based stability analysis is effective for studying delay differential systems with time-periodic coefficients and periodically varying delays. The stability analysis also helps bring forth the benefits of variable spindle speed milling operations compared to constant spindle speed milling operations.

**Keywords:** Constant spindle speed milling · Delay differential equations · Floquet analysis · Semidiscretization method · Stability · Time periodic delays · Variable spindle speed milling

### 5.1 Introduction

Time delays have been used to develop models of many practical systems where the rate of change of the state depends not only on the current states but also on the past states. The mathematical model of a system with a time lag can be formulated as a set of delay differential equations (DDEs). As also discussed in other parts of this book, time-delay systems have been used to model manufacturing processes, neural networks, mechanical systems with viscoelasticity, nuclear reactors, distributed

networks, internal combustion engines, species interaction in microbiology, learning, epidemiology, and physiology [1]. These practical applications have spurred a considerable number of studies on the dynamics of systems with time delays. In the context of stability, considerable research has been carried out on systems with constant delays [2–5]. For a system with constant coefficients and a constant delay, one can determine the sufficient conditions for the stability and the instability of the system by using the method of Lyapunov functionals [6]. However, due to the nature of the Lyapunov function, this stability analysis is limited for practical problems. Frequency-domain methods provide an alternative for the stability analysis of systems with constant time delays and constant coefficients [2, 3].

As in the case of a system without a delay, an equilibrium point of DDEs is asymptotically stable if and only if all the roots ( $\lambda_i, i = 1, 2, \dots$ ) of the characteristic equation have negative real parts. The stability of periodic solutions of ordinary differential equations can be determined by using Floquet theory: if all the eigenvalues of the Floquet transition matrix (or also called monodromy matrix) are inside the unit circle of the complex plane, then the system is asymptotically stable. Floquet theory can be extended to DDEs with time-periodic coefficients [7, 8]. Through this extension, one can define a monodromy operator  $U_T$  that can be viewed as an infinite-dimensional Floquet transition matrix and determine the stability of a periodic solution of the time-delay system by investigating the infinite spectrum of eigenvalues. Recently, numerical methods have been proposed to approximate the infinite dimensional monodromy operator with a finite dimensional one. Examples of numerical methods based on Floquet theory include the methods based on Chebyshev polynomials [9], temporal finite element method [10–12], full discretization method [13], averaged coefficients method [4, 14], and semidiscretization method [15–17].

Compared to the types of systems considered thus far, the situation is more complex when the system has time-varying delays. Such delays can be potentially disastrous in terms of stability and oscillations in some systems, while also being beneficial in other systems; for example, metal cutting process [18], neural networks [19], and biology [20, 21]. As a representative form of such systems, the following form of linear DDEs with periodically varying time delays and periodic coefficients is considered:

$$\dot{\hat{\mathbf{X}}}(t) = \mathbf{A}(t)\hat{\mathbf{X}}(t) + \sum_{j=1}^n \mathbf{B}_j(t)\hat{\mathbf{X}}(t - \tau_j(t)) + \mathbf{F}(t). \quad (5.1)$$

Here,  $\hat{\mathbf{X}} \in R^n$ ,  $\mathbf{A}(t)$  and  $\mathbf{B}_j(t)$  are  $n \times n$  matrices with period  $T_0$ ,  $\tau_j(t)$  is the time-varying delay with period  $T_j$ , and  $\mathbf{F}(t)$  is a forcing vector with period  $T_0$ . It is assumed that the periods  $T_0$  and  $T_1, T_2, \dots, T_n$  are commensurate. If otherwise, the system is quasiperiodic, and the Floquet analysis does not apply. In the following analysis, the least period of the system (i.e., the least multiple of  $T_0, T_1, \dots, T_n$ ) is denoted by  $T$ . In system (5.1), there are two different types of time effects, one due to the time-varying coefficient matrices and another due to the time-varying delay. Thus, this system is different from a system either with only time-varying coefficient matrices and a constant time delay or with constant coefficient matrices and a periodically varying delay. Hence, system (5.1) requires a stability analysis

approach different from what has been used for other systems. Liu and Liao [19], Jiang et al. [22], and Zhou et al. [23] determined sufficient conditions for the stability of the periodic solutions of cellular neural networks and bidirectional associative memory neural networks with time-varying delays by constructing suitable Lyapunov functionals. Schely and Gourley [21] considered the effect of daily, seasonal, or annual fluctuations on the population and obtained system models with periodically perturbed delays. The two-timing method was used to investigate how the stability of the system differs from that with a constant delay.

Apart from the use of DDEs with time-periodic delays to model neural networks and systems in biology, the dynamics of a milling process can also be described by DDEs with time-periodic delays and -periodic coefficients. In particular, such systems can be used to model a variable spindle speed (VSS) milling process or a constant spindle speed (CSS) milling process with consideration of feed-rate effect on the delay. Altintas and Chan [24] and Radulescu et al. [25, 26] investigated the stability of a VSS milling process with periodic time-varying delay through time-domain simulations. When compared to purely simulation driven schemes, numerical schemes with an analytical basis can provide faster and reliable stability predictions. Tsao et al. [27] used the angular position as an independent variable instead of time, and they employed a full discretization scheme to analyze the stability of the resulting system. Sastry et al. [28] directly analyzed the stability of a milling process with a sinusoidal spindle speed variation by using the full discretization scheme. Yilmaz et al. [13] also used the full discretization scheme to analyze the stability of a milling process with random spindle speed variations. Sastry et al. [29] used Floquet theory to carry out stability analysis for a variable speed face-milling process. Long and Balachandran [30] modeled a VSS milling process by using an inhomogeneous set of DDEs, and they used the semidiscretization method to analyze the stability of periodic solutions of these DDEs with time-varying coefficients and a time-varying delay. The material presented in this chapter complements the discretization methods discussed in chap. 4.

The rest of this chapter is organized as follows. First, the stability analysis of periodic orbits and the tools available for carrying out this analysis are briefly discussed. Following that, the construction of the approximate finite dimensional monodromy matrix by using the semidiscretization method is presented. Then, different examples of systems with constant and periodically varying delays are considered and the application of the semidiscretization method is illustrated. One of the detailed examples is on milling dynamics. Finally, some remarks are collected together at the end.

## 5.2 Stability Analysis of Systems with Periodically Varying Delays

As discussed in [21], periodic delays can have either a stabilizing effect or a destabilizing effect. There are many criteria and methods that can be used to determine the stability of a periodic solution of a delay differential system. The

Lyapunov–Krasovskii functional-based method is one of them. Liu and Liao [19], Jiang et al. [22], and Zhou et al. [23] used this method to determine sufficient conditions for the stability of periodic solutions of neural networks with time-varying delays. Krasovskii [31] is the first one who realized that such a “Lyapunov functional” should depend on the delayed state. The use of Lyapunov functional methods is limited because of the difficulty in constructing these functionals. The frequency-domain method is convenient to use for the stability analysis of linear time-invariant systems with delays.

Here, for the linear periodic delayed systems of the form of (5.1), the extended Floquet theory [7, 8] is used to determine the stability. Let the nominal periodic orbit of system (5.1) be represented by  $\mathbf{X}_0(t)$ . Then, a perturbation  $\mathbf{X}(t)$  is provided to this nominal orbit resulting in

$$\dot{\mathbf{X}}(t) = \mathbf{X}_0(t) + \mathbf{X}(t). \quad (5.2)$$

After substituting (5.2) into (5.1), the equation governing the perturbation is determined as

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t) + \sum_{j=1}^n \mathbf{B}_j(t)\mathbf{X}(t - \tau_j(t)). \quad (5.3)$$

Let  $\tau_{\max} = \max(\tau_j(t))$  and  $C$  denote the space of continuous functions from  $[-\tau_{\max}, 0]$  to  $R^n$ , with the norm in  $C$  given by  $\|\phi\| = \max |\phi(t)|$  for  $-\tau_{\max} \leq t \leq 0$ . The function  $\mathbf{X}_t \in C$  is defined as  $\mathbf{X}_t(\theta) = \mathbf{X}(t + \theta)$  with  $t \in [-\tau_{\max}, 0]$ . For any  $s \in R$  and  $\phi \in C$ , there is a unique solution  $\mathbf{X}(\cdot, s, \phi)$  of (5.3) defined on  $[s, \infty)$ , such that the associated function  $\mathbf{X}_t(\cdot, s, \phi)$  defined as  $\mathbf{X}_t(\theta, s, \phi) = \mathbf{X}(t + \theta, s, \phi)$ ,  $t \in [-\tau_{\max}, 0]$  is continuous in  $t$ ,  $s$ , and  $\phi$ , and it satisfies the initial condition  $\mathbf{X}_s(\cdot, s, \phi) = \phi$ . Here,  $s$  is the initial instant and  $\phi$  is the initial function. One can define the solution operator  $U : C \rightarrow C$  as

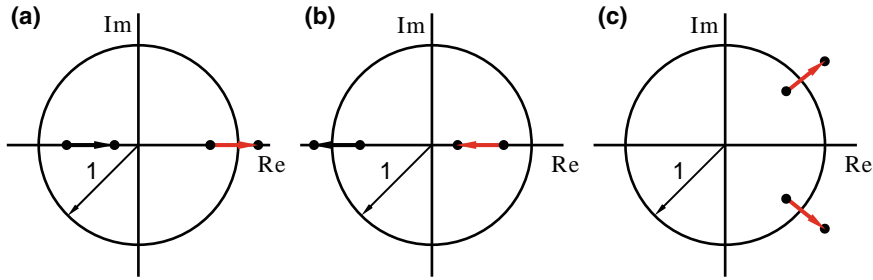
$$\mathbf{X}_t(\cdot, s, \phi) = U(t, s)\phi. \quad (5.4)$$

Without loss of generality, the initial instant can be set to  $s = 0$ . Considering the periodicity of (5.3), the monodromy operator can be defined by setting  $t = T$

$$\mathbf{X}_T(\cdot, 0, \phi) = U_T\phi, \quad (5.5)$$

where  $U_T := U(T, 0)$  is a short notation for the monodromy operator associated with the initial instant  $s = 0$ . The spectrum of the monodromy operator, denoted by  $\sigma(U_T)$ , is at most countable, a compact set of the complex plane with the only possible limit point being zero. If  $\mu \in \sigma(U_T)$  and  $\mu \neq 0$ , then there exists an initial function  $\phi \in C$  such that  $U_T\phi = \mu\phi$ . The nonzero elements of  $\sigma(U_T)$  are called characteristic multipliers or Floquet multipliers of the system. If all of the Floquet multipliers are within the unit circle of the complex plane, then the corresponding periodic solution of (5.1) is stable. If one or more of the Floquet multipliers are on the unit circle, while the rest of them are inside the unit circle, the corresponding periodic solution may undergo a bifurcation [32].

In Fig. 5.1, three possible ways for the loss of stability of a periodic solution are shown. When a real Floquet multiplier crosses the unit circle at  $+1$ , then the



**Fig. 5.1** Scenarios showing how the Floquet multipliers leave the unit circle for different bifurcations: (a) cyclic-fold bifurcation, (b) period-doubling (flip) bifurcation, and (c) secondary Hopf bifurcation

corresponding bifurcation is called a cyclic-fold bifurcation. On the other hand, when a real Floquet multiplier crosses the unit circle at  $-1$ , the corresponding bifurcation is called a period-doubling or flip bifurcation. If a pair of complex Floquet multipliers leaves the unit circle, then the corresponding bifurcation is called a secondary Hopf bifurcation. For time-periodic delay differential systems such as (5.3), it is difficult to determine the monodromy matrix, which has no closed-form solutions. In practical applications, a finite dimensional monodromy matrix is constructed to approximate the infinite dimensional monodromy operator. In Sect. 5.3, the semidiscretization method is presented to construct an approximate monodromy matrix for system (5.3).

### 5.3 Approximation of the Monodromy Matrix by using the Semidiscretization Method

In this section, the steps underlying the semidiscretization method are shown for constructing a finite dimensional approximation to the infinite dimensional monodromy operator. The eigenvalues of the approximate monodromy matrix are used to examine the local stability of the considered periodic solution.

As a first step, the time period  $T$  of the periodic orbit is divided into  $K$  equal intervals of length  $\Delta t = T/K$  and the discrete time scale  $t_i = i\Delta t$  is introduced. For  $t \in [t_i, t_{i+1}]$ , the time-periodic coefficient matrices in (5.3) are approximated as

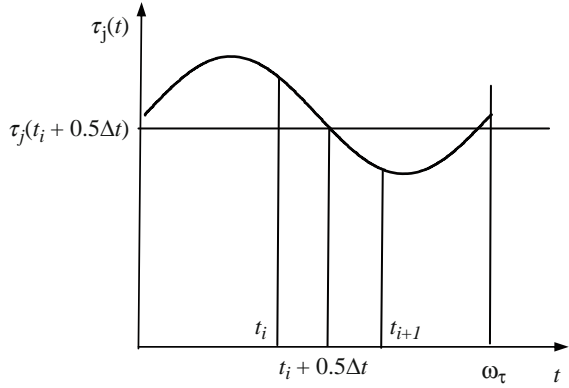
$$\mathbf{A}_i \approx \frac{1}{\Delta t} \int_{t_i}^{t_{i+1}} \mathbf{A}(s) ds, \quad (5.6)$$

$$\mathbf{B}_{j,i} \approx \frac{1}{\Delta t} \int_{t_i}^{t_{i+1}} \mathbf{B}_j(s) ds, \quad (5.7)$$

and the time delays are approximated as

$$\tau_{j,i} = \tau_j(t_i + 0.5\Delta t), \quad (5.8)$$

**Fig. 5.2** Approximation of the time-varying delay



as shown in Fig. 5.2. The relationship between  $\Delta t$  and the time delay  $\tau_{j,i}$  is given by

$$\tau_{j,i} = (l_{j,i} + l_{r,j,i} + 0.5)\Delta t, \quad (5.9)$$

where

$$l_{r,j,i} = \text{mod}(\tau_{j,i} - 0.5\Delta t, \Delta t) \quad (5.10)$$

and

$$l_{j,i} = \frac{\tau_{j,i}}{\Delta t} - l_{r,j,i} - 0.5. \quad (5.11)$$

The delayed state is then approximated as

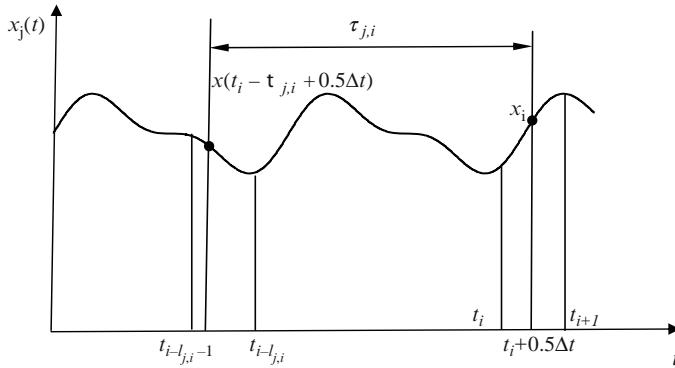
$$\begin{aligned} \mathbf{X}(t - \tau_j(t)) &\approx \mathbf{X}(t_i + 1/2\Delta t - \tau_{j,i}) \\ &\approx (1 - l_{r,j,i})\mathbf{X}(t_{i-l_{j,i}}) + l_{r,j,i}\mathbf{X}(t_{i-l_{j,i}-1}). \end{aligned} \quad (5.12)$$

The sketch of this approximation is provided in Fig. 5.3. Then, over each time interval  $t \in [t_i, t_{i+1}]$  for  $i = 0, 1, 2, \dots, N-1$ , system (5.3) can be approximated as

$$\dot{\mathbf{X}}(t) = \mathbf{A}_i \mathbf{X}(t) + \sum_{j=1}^n \mathbf{B}_{j,i} [(1 - l_{r,j,i})\mathbf{X}(t_{i-l_{j,i}}) + l_{r,j,i}\mathbf{X}(t_{i-l_{j,i}-1})], \quad t \in [t_i, t_{i+1}]. \quad (5.13)$$

Thus, the infinite-dimensional system (5.3) has been approximated by a series of autonomous, ordinary differential equations in each discretization interval. It is noted that the autonomous system has a constant excitation or forcing term that arises due to the delay effect in each interval. To proceed further, the matrix  $\mathbf{A}_i$  is assumed to be invertible for all  $i$ . Then, for  $t \in [t_i, t_{i+1}]$ , the solution of (5.13) takes the form

$$\begin{aligned} \mathbf{X}(t) &= e^{\mathbf{A}_i(t-t_i)} \mathbf{X}_i \\ &+ (e^{\mathbf{A}_i(t-t_i)} - \mathbf{I}) \sum_{j=1}^n \mathbf{A}_i^{-1} \mathbf{B}_{j,i} [(1 - l_{r,j,i})\mathbf{X}(t_{j-l_{j,i}}) + l_{r,j,i}\mathbf{X}(t_{j-l_{j,i}-1})], \end{aligned} \quad (5.14)$$



**Fig. 5.3** Approximation of the delayed term

where  $\mathbf{X}_i = \mathbf{X}(t_i)$  is used for compact notation. When  $t = t_{i+1}$ , the system (5.14) leads to

$$\mathbf{X}_{i+1} = \mathbf{M}_{i,0}\mathbf{X}_i + \sum_{j=1}^n (\mathbf{M}_{j,l_{j,i}}\mathbf{X}_{i-l_{j,i}} + \mathbf{M}_{j,l_{j,i}+1}\mathbf{X}_{i-l_{j,i}-1}), \quad (5.15)$$

where the associated matrices are given by

$$\mathbf{M}_{i,0} = e^{\mathbf{A}_i \Delta t}, \quad (5.16)$$

$$\mathbf{M}_{j,l_{j,i}} = (e^{\mathbf{A}_i \Delta t} - \mathbf{I})\mathbf{A}_i^{-1}\mathbf{B}_{j,i}(1 - l_{r_{j,i}}), \quad (5.17)$$

and

$$\mathbf{M}_{j,l_{j,i}+1} = (e^{\mathbf{A}_i \Delta t} - \mathbf{I})\mathbf{A}_i^{-1}\mathbf{B}_{j,i}l_{r_{j,i}}. \quad (5.18)$$

Next, the augmented state vector

$$\mathbf{Y}_i = (\mathbf{X}_i, \mathbf{X}_{i-1}, \dots, \mathbf{X}_{i-l_{\max}})^T, \quad (5.19)$$

where

$$l_{\max} = \frac{\tau_{\max}}{\Delta t} - \text{mod} \frac{\tau_{\max}}{\Delta t} + 1 \quad (5.20)$$

is introduced. Combining (5.15)–(5.19), one can construct the linear discrete map

$$\mathbf{Y}_{i+1} = \mathbf{D}_i \mathbf{Y}_i, \quad (5.21)$$

where the matrix  $\mathbf{D}_i$  is given by

$$\mathbf{D}_i = \begin{bmatrix} \mathbf{M}_{i,0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & \vdots & & \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix} + \sum_{j=1}^n \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{M}_{j,l_{j,i}} & \mathbf{M}_{j,l_{j,i}+1} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & & \vdots & \vdots & & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}. \quad (5.22)$$

Here, the second subscript of submatrices  $\mathbf{M}_{j,l_{j,i}}$  and  $\mathbf{M}_{j,l_{j,i}+1}$  refers to the corresponding location in  $\mathbf{D}_i$ .

Consecutive applications of the discrete map (5.21) over a period  $T = K\Delta t$  results in

$$\mathbf{Y}_K = \mathbf{D}_{K-1} \cdots \mathbf{D}_1 \mathbf{D}_0 \mathbf{Y}_0, \quad (5.23)$$

where the transition matrix can be identified as

$$\Phi = \mathbf{D}_{K-1} \cdots \mathbf{D}_1 \mathbf{D}_0. \quad (5.24)$$

This matrix  $\Phi$  represents a finite-dimensional approximation of the monodromy operator associated with the periodic orbit  $\mathbf{X}_0(t)$  of (5.1) and the trivial solution  $\mathbf{X}(t) = \mathbf{0}$  of (5.3). If the eigenvalues of this matrix are all within the unit circle, then the trivial fixed point of (5.3) is stable, and hence, the associated periodic orbit of (5.1) is stable. At a bifurcation point, one or more of the eigenvalues of the transition matrix are on the unit circle of the complex plane as illustrated in Fig. 5.1.

## 5.4 Applications

In this section, different DDEs are analyzed by using the semidiscretization method. First, a scalar DDE is considered with a periodic coefficient and a single delay is considered, followed by an autonomous scalar system with two constant delays. Then, the damped delayed Mathieu equation with a periodic coefficient in the delay term is analyzed. For each of these systems, it is shown as to how the stable and unstable regions can be identified in the considered parameter space. Finally, CSS milling and VSS milling are considered, and the associated stability diagrams are constructed and discussed.

### 5.4.1 Scalar Delay Differential Equation with Periodic Coefficient and One Delay

As a special case of (5.3), consider the scalar differential system with a constant coefficient, a periodic coefficient, and a constant delay term

$$\dot{x}(t) = (a + \varepsilon \cos(\omega t))x(t) + bx(t - \tau), \quad (5.25)$$

where  $a$ ,  $\varepsilon$ , and  $b$  are scalar parameters;  $T = 2\pi/\omega$  is the period of the system; and  $\tau$  is the time delay. This system (5.25) has a parametric excitation term, and in one of the later examples, the delayed term also has a parametric excitation effect. The solution of system (5.25) and the associated monodromy operator cannot be constructed in closed form. An approximation for this monodromy operator

is obtained by using the semidiscretization method and used to construct the stability charts for this system. Here, for convenience, the authors fix the time period to  $T = 2$  and the time delay to  $\tau = 1$ , and construct stability charts in the  $a - b$  plane for different values of  $\varepsilon$ .

First, the authors divide the period  $T$  into  $K = 20$  equal intervals of length  $\Delta t = T/K = 0.1$  and introduce the discrete time scale  $t_i = i\Delta t$ . For  $t \in [t_i, t_{i+1}]$ , the periodic coefficient is approximated by the piecewise constant values

$$a_i \approx \int_{t_j}^{t_{i+1}} a + \varepsilon \cos(\omega s) ds \quad (5.26)$$

and the delayed term is approximated as

$$x(t - \tau) \approx x(t_i + 0.5 \Delta t - \tau) \approx (1 - \text{lr})x(t_{i-1}) + \text{lr}x(t_{i-1-1}), \quad (5.27)$$

where

$$\text{lr} = \text{mod}(\tau - 0.5 \Delta t, \Delta t) = \text{mod}(1 - 0.05, 0.1) = 0.5 \quad (5.28)$$

and

$$l = \frac{\tau}{\Delta t} - \text{lr} - 0.5 = \frac{1}{0.1} - 0.5 - 0.5 = 9. \quad (5.29)$$

Then, over each time interval  $t \in [t_i, t_{i+1}]$  for  $i = 0, 1, 3, \dots, K - 1$ , system (5.25) can be approximated as

$$\dot{x}(t) = a_i x(t) + (1 - \text{lr})x(t_{i-1}) + \text{lr}x(t_{i-1-1}). \quad (5.30)$$

On substitution of the values for  $\text{lr}$  and  $l$ , one obtains

$$\dot{x}(t) = a_i x(t) + 0.5x(t_{i-9}) + 0.5x(t_{i-10}). \quad (5.31)$$

For  $t \in [t_i, t_{i+1}]$ , the solution of (5.31) takes the form

$$x(t) = e^{a_i(t-t_i)}x_i + 0.5a_i^{-1}b(e^{a_i(t-t_i)} - 1)[x_{i-9} + x_{i-10}]. \quad (5.32)$$

When  $t = t_{i+1}$ , system (5.32) leads to

$$x_{i+1} = x(t_{i+1}) = e^{a \Delta t}x_i + 0.5a^{-1}b(e^{a \Delta t} - 1)(x_{i-9} + x_{i-10}). \quad (5.33)$$

After introducing the augmented state vector

$$\mathbf{Y}_i = (x_i, x_{i-1}, \dots, x_{i-10})^T, \quad (5.34)$$

and combining (5.33) and (5.34), one can construct the linear discrete map

$$\mathbf{Y}_{i+1} = \mathbf{D}_i \mathbf{Y}_i, \quad (5.35)$$

where the matrix  $\mathbf{D}_i$  is given by

$$\mathbf{D}_i = \begin{bmatrix} e^{a_i \Delta t} & 0 & \cdots & 0 & m_i & m_i \\ 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & & \ddots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix} \quad (5.36)$$

with  $m_i = 0.5a_i^{-1}b(e^{a_i \Delta t} - 1)$ .

After  $k$  consecutive applications of (5.35), one ends up with

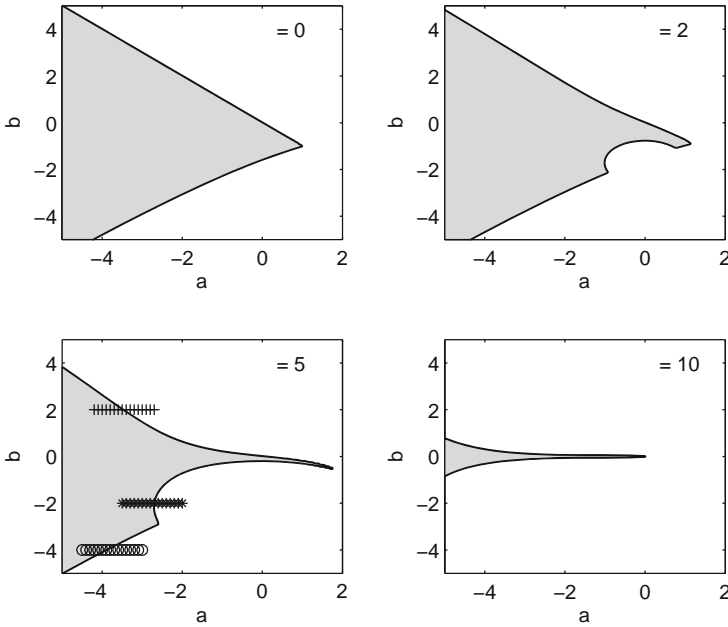
$$\mathbf{Y}_K = \mathbf{D}_{K-1} \cdots \mathbf{D}_1 \mathbf{D}_0 \mathbf{Y}_0, \quad (5.37)$$

from which the transition matrix can be identified as

$$\Phi = \mathbf{D}_{K-1} \cdots \mathbf{D}_1 \mathbf{D}_0. \quad (5.38)$$

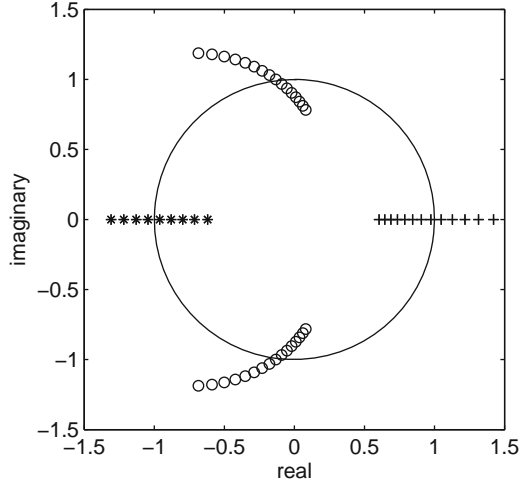
Here,  $k = 20$ .

Equation (5.37) provides an approximate discrete map of system (5.25). One can determine the stability of the system by checking the eigenvalues of the monodromy matrix  $\Phi$ . The boundaries between stable and unstable regions in the parameter space of  $a$  and  $b$  for different  $\varepsilon$  are shown in Fig. 5.4. The three possible routes for



**Fig. 5.4** Stability charts for system (5.25) obtained by semidiscretization method for  $T = 2$ ,  $\tau = 1$ , and  $K = 20$ . Stable regions are shaded and unstable regions are not shaded

**Fig. 5.5** The trajectories of Floquet multipliers in the complex plane for  $b = -4, -2$ , and  $2$ . The markers  $\circ$ ,  $*$ , and  $+$  correspond to secondary Hopf, period-doubling, and cyclic-fold bifurcations, respectively



stability loss is demonstrated for the case  $\varepsilon = 5$ . The parameter  $b$  is fixed at  $-4$  ( $\circ$ ),  $-2$  ( $*$ ) and  $2$  ( $+$ ), while the parameter  $a$  is increased. The corresponding locations of the critical Floquet multipliers are shown in Fig. 5.5. For the case  $b = -4$  ( $\circ$ ), a secondary Hopf bifurcation occurs. For the case  $b = -2$  ( $*$ ), a period doubling (flip) bifurcation occurs, and for the case  $b = 2$  ( $+$ ), a cyclic-fold bifurcation occurs.

### 5.4.2 Scalar Autonomous Delay Differential Equation with Two Delays

In this section, the authors consider a DDE with two delays. Such a system is used to describe the dynamics of many biological and physical systems involving feedback incorporated delays in their action. The equation of interest is

$$\dot{x}(t) = ax(t) + bx(t - \tau_1) + cx(t - \tau_2). \quad (5.39)$$

This system is autonomous; that is, it does not contain any time-periodic coefficients or terms, and the time delays are also constants. Therefore, the period  $T$  and the length of the discretization interval  $\Delta t = T/K$  can arbitrarily be chosen. Let  $\Delta t = 2\pi/K$  and the discrete time scale be  $t_i = i\Delta t$ . For  $t \in [t_i, t_{i+1}]$ , the delay terms are approximated as

$$\begin{aligned} x(t - \tau_1) &\approx x(t_i + 0.5 \Delta t - \tau_1) \approx (1 - l_{r1})x_{i+1-l_1} + l_{r1}x_{i-l_1} \\ x(t - \tau_2) &\approx x(t_i + 0.5 \Delta t - \tau_2) \approx (1 - l_{r2})x_{i+1-l_2} + l_{r2}x_{i-l_2} \end{aligned} \quad (5.40)$$

where  $l_{r1} = \text{mod}(\tau_1 - 0.5 \Delta t, \Delta t)$ ,  $l_{r2} = \text{mod}(\tau_2 - 0.5 \Delta t, \Delta t)$ ,  $l_1 = \tau_1/\Delta t - 0.5 - l_{r1}$ , and  $l_2 = \tau_2/\Delta t - 0.5 - l_{r2}$ .

Similar to the system with one delay treated previously, one can construct the discrete map

$$\mathbf{Y}_{i+1} = \mathbf{D}_i \mathbf{Y}_i, \quad (5.41)$$

where the matrix  $\mathbf{D}_i$  is given by

$$\mathbf{D}_i = \begin{bmatrix} e^{a \Delta t} & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & & \ddots & & & \\ 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix} + \sum_{j=1}^2 \begin{bmatrix} 0 & \cdots & M_{j,l_j} & M_{j,l_j+1} & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (5.42)$$

with

$$\begin{aligned} M_{1,l_1} &= (1 - l r_1) \cdot a^{-1} b (e^{a \Delta t} - 1), \\ M_{1,l_1+1} &= l r_1 \cdot a^{-1} b (e^{a \Delta t} - 1), \\ M_{2,l_2} &= (1 - l r_2) \cdot a^{-1} c (e^{a \Delta t} - 1), \\ M_{2,l_2+1} &= l r_2 \cdot a^{-1} c (e^{a \Delta t} - 1). \end{aligned}$$

In each discrete step, the transition matrix  $\mathbf{D}_i$  is the same as given above (the elements of  $\mathbf{D}_i$  do not depend on  $i$ ). Hence, the stability information can be obtained from an analysis of the eigenvalues of matrix  $\Phi = \mathbf{D}_i$ .

The stability of system (5.39) can be determined by investigating the eigenvalues of the transition matrix  $\Phi$ . The corresponding stability charts in the parameter space of  $\tau_1$  and  $\tau_2$  are presented in Fig. 5.6 for different values of the coefficients  $a, b$ , and  $c$ .

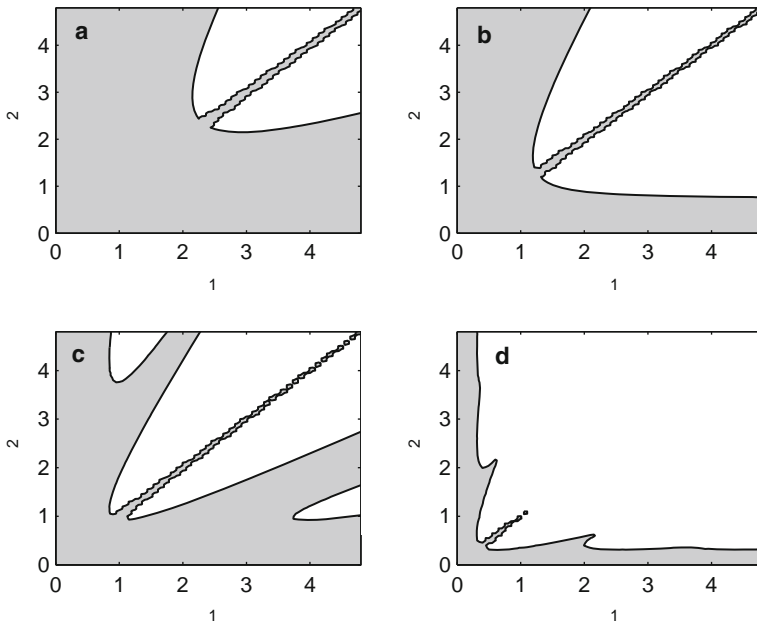
### 5.4.3 Damped and Delayed Mathieu Equation

In order to investigate the combined effect of parametric excitation and a single time delay, the damped and delayed Mathieu equation has been studied for different parameter combinations in [16, 33]. Here, the authors consider the case, where the coefficient of the delayed term has a time-periodic variation. The equation of interest is

$$\ddot{x}(t) + c\dot{x}(t) + (a + \varepsilon_1 \cos(\omega t))x(t) = (b + \varepsilon_2 \cos(\omega t))x(t - \tau), \quad (5.43)$$

where the  $c$  is the damping coefficient,  $\tau$  is the constant time delay, and the time period is  $T = 2\pi/\omega$ .

The system (5.43) can be rewritten in the state-space form



**Fig. 5.6** Stability chart of system (5.39) obtained by semidiscretization method: (a)  $a = b = c = 0.5$ , (b)  $a = c = 0.5, b = 1$ , (c)  $a = c = 2, b = 1$ , and (d)  $b = c = 2, a = 1$ . Stable regions are shaded and unstable regions are not shaded

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{B}(t)\mathbf{X}(t - \tau) \quad (5.44)$$

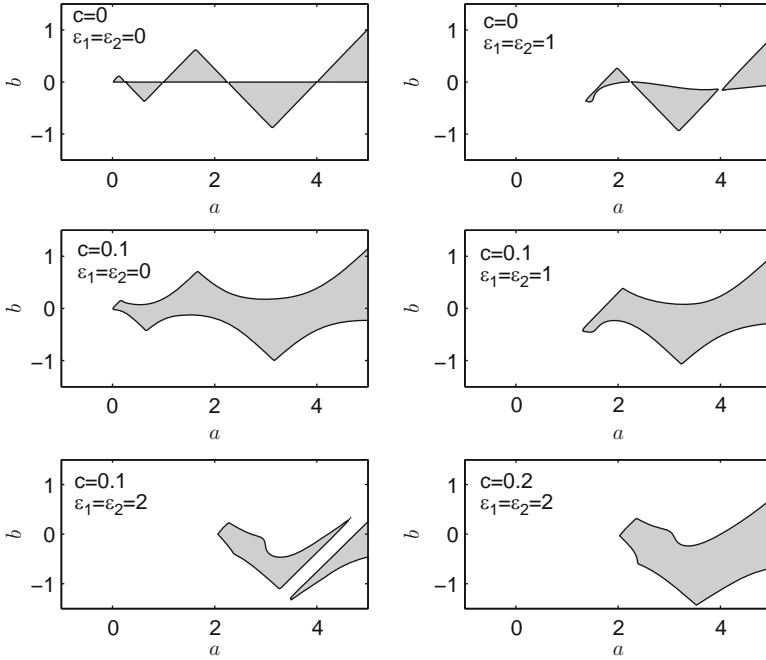
with

$$\mathbf{X} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}, \quad \mathbf{A}(t) = \begin{bmatrix} 0 & 1 \\ -(a + \varepsilon_1 \cos(\omega t)) & -c \end{bmatrix}, \quad (5.45)$$

$$\mathbf{B}(t) = \begin{bmatrix} 0 & 0 \\ b + \varepsilon_2 \cos(\omega t) & 0 \end{bmatrix} \quad (5.46)$$

The application of the semidiscretization method to system (5.44) is similar to the autonomous DDE with one delay. The coefficient matrices  $\mathbf{A}(t)$  and  $\mathbf{B}(t)$  are approximated according to (5.7), and the procedure presented in Sect. 5.3 can be followed to construct the Floquet transition matrix  $\Phi$ . Then, the solution stability can be determined by analyzing the eigenvalues of  $\Phi$ . The stability charts are presented in Fig. 5.7 for  $T = \tau = 2\pi$  and different values of the coefficients  $c, \varepsilon_1$ , and  $\varepsilon_2$ . It is noted that  $c = 0$  corresponds to an undamped system and  $c > 0$  corresponds to a damped system. For fixed  $\varepsilon_1$  and  $\varepsilon_2$ , the region of stability increases with increase in damping.

In the first three subsections of Sect. 5.4, scalar autonomous, scalar nonautonomous, and a two-dimensional nonautonomous delay differential systems have been considered and the use of the semidiscretization method to determine the stability domains in each case has been illustrated. Next, two systems that arise in

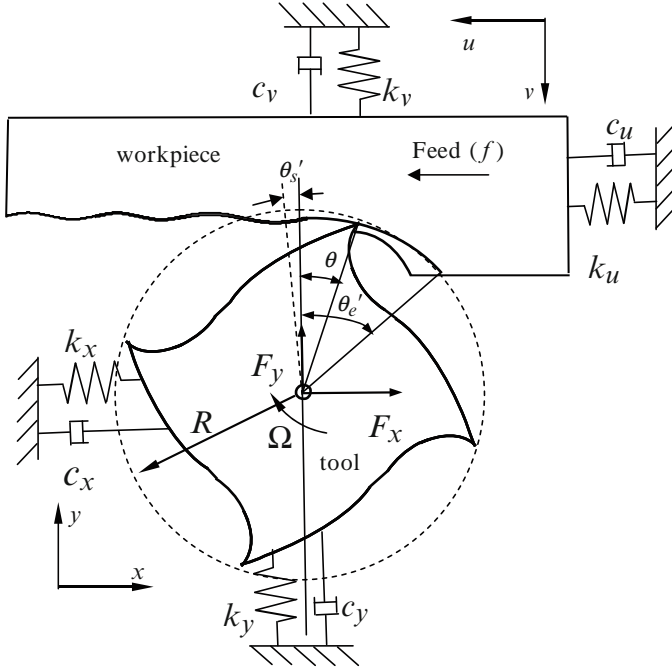


**Fig. 5.7** Stability chart of system (5.43) obtained by semidiscretization method with  $\tau = T = 2\pi$ . Stable regions are shaded and unstable regions are not shaded

the context of milling dynamics are considered. These systems are comparatively more complex than the previous three systems considered, but the application of the semidiscretization method follows along the lines of Sect. 5.3.

#### 5.4.4 CSS Milling Process: Nonautonomous DDE with Time-Periodic Delays

In this section, the model developed by Long and Balachandran [17] is briefly revisited and the stability analysis, carried out for a CSS milling process based on the semidiscretization method, is presented. In Fig. 5.8, a multi-degree-of-freedom configuration representative of a workpiece–tool system is illustrated for milling operations with a cylindrical end mill. For convenience, the  $X$ -direction is oriented along the feed direction of the cutter. The vertical axis of the tool is oriented along the  $Z$ -direction. The spindle rotational speed in  $\text{rad s}^{-1}$  is represented by  $\Omega$  and the angular position is represented by  $\theta$ . The quantities  $\theta_s$  and  $\theta_e$  represent the entry cutting angle and exit cutting angle, respectively, and these angles define the static cutting zone. The forces  $F_x$  and  $F_y$  act on the cutter, and the forces  $F_u$  and  $F_v$  act on the workpiece. The natural frequencies associated with the torsion modes and the  $Z$ -direction vibration modes are expected to be higher than those associated with



**Fig. 5.8** Workpiece-tool system model

the primary bending vibration modes along the  $X$ -direction and the  $Y$ -direction. For this reason, only the vibration modes in the horizontal plane are considered here. In developing these models, the modal properties of the tool and the workpiece are assumed to be known, and hence, a system with a flexible tool and a flexible workpiece can be represented by an equivalent lumped parameter system.

The governing equations are of the form

$$\begin{aligned}
 m_x \ddot{q}_x(t) + c_x \dot{q}_x(t) + k_x q_x(t) &= F_x(t; \tau(t, j, z)), \\
 m_y \ddot{q}_y(t) + c_y \dot{q}_y(t) + k_y q_y(t) &= F_y(t; \tau(t, j, z)), \\
 m_u \ddot{q}_u(t) + c_u \dot{q}_u(t) + k_u q_u(t) &= F_u(t; \tau(t, j, z)), \\
 m_v \ddot{q}_v(t) + c_v \dot{q}_v(t) + k_v q_v(t) &= F_v(t; \tau(t, j, z)),
 \end{aligned} \tag{5.47}$$

where both the tool and the workpiece have two degrees of freedom. The variables  $q_x$  and  $q_y$ , respectively, represent the tool dynamic displacements measured along the  $X$  and the  $Y$  directions in a reference frame, whose origin is located on the tool center and shares the rigid-body translation of the tool due to a constant feed rate. The variables  $q_u$  and  $q_v$  represent the workpiece displacements measured, respectively, along the  $U$  and  $V$  directions in a fixed reference frame. The quantities  $m_x, m_y, m_u, m_v$  are the modal masses, the quantities  $c_x, c_y, c_u, c_v$  are the modal damping coefficients, and the quantities  $k_x, k_y, k_u, k_v$  are the modal stiffness parameters associated with the motions along the  $X, Y, U$ , and  $V$  directions,

respectively. The cutting force components, which appear on the right-hand side of the equations, are time-periodic functions. Furthermore, considering the feed motion effect, a variable time delay is introduced in the governing equations through the cutting-force components. A variable time delay is associated with each cutting tooth and the number of variable time delays is equal to the number of flutes. This variable time delay depends on the feed rate, the radius of the tool, the spatial location along the Z-direction, and the spindle rotation speed.

System (5.47) can be put into the state-space form

$$\dot{\mathbf{Q}}(t) = \mathbf{A}(t)\mathbf{Q}(t) + \sum_{j=1}^N \mathbf{B}_j(t)\mathbf{Q}(t - \tau_j(t)) + \mathbf{F}(t), \quad (5.48)$$

where  $\mathbf{Q} = \{q_x \ q_y \ q_u \ q_v \ \dot{q}_x \ \dot{q}_y \ \dot{q}_u \ \dot{q}_v\}^T$ ,  $\mathbf{A}(t)$  is the coefficient matrix for the vector of present states

$$\mathbf{A}(t) = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(\mathbf{K} - \hat{\mathbf{K}}(t)) & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \quad (5.49)$$

and  $\mathbf{B}_j(t)$  is the coefficient matrix associated with the vectors of delayed states. These matrices are given by

$$\mathbf{B}_j(t) = \int_{z_1(t,j)}^{z_2(t,j)} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}\hat{\mathbf{K}}^j(t, z) & \mathbf{0} \end{bmatrix} dz, \quad (5.50)$$

the last term associated with the inhomogeneous term is

$$\mathbf{F}(t) = \begin{bmatrix} \mathbf{0} \\ \bar{\mathbf{K}}(t)f \end{bmatrix}, \quad (5.51)$$

where  $\mathbf{M}$ ,  $\mathbf{C}$ , and  $\mathbf{K}$  are the mass matrix, the damping matrix, and the stiffness matrix, respectively.  $\hat{\mathbf{K}}(t)$ ,  $\hat{\mathbf{K}}^j(t, z)$ , and  $\bar{\mathbf{K}}(t)$  are periodic coefficient matrices associated with tool pass excitation effect and they take the following forms:

$$\hat{\mathbf{K}}(t) = \sum_{j=1}^N \int_{z_1(t,j)}^{z_2(t,j)} \begin{bmatrix} k_{11}^j(t, z) & k_{12}^j(t, z) & k_{11}^j(t, z) & k_{12}^j(t, z) \\ k_{21}^j(t, z) & k_{22}^j(t, z) & k_{21}^j(t, z) & k_{22}^j(t, z) \\ k_{11}^j(t, z) & k_{12}^j(t, z) & k_{11}^j(t, z) & k_{12}^j(t, z) \\ k_{21}^j(t, z) & k_{22}^j(t, z) & k_{21}^j(t, z) & k_{22}^j(t, z) \end{bmatrix} dz, \quad (5.52)$$

$$\hat{\mathbf{K}}^j(t, z) = \begin{bmatrix} k_{11}^j(t, z) & k_{12}^j(t, z) & k_{11}^j(t, z) & k_{12}^j(t, z) \\ k_{21}^j(t, z) & k_{22}^j(t, z) & k_{21}^j(t, z) & k_{22}^j(t, z) \\ k_{11}^j(t, z) & k_{12}^j(t, z) & k_{11}^j(t, z) & k_{12}^j(t, z) \\ k_{21}^j(t, z) & k_{22}^j(t, z) & k_{21}^j(t, z) & k_{22}^j(t, z) \end{bmatrix}, \quad (5.53)$$

$$\bar{\mathbf{K}}(t) = \begin{bmatrix} k_{11}^j(t, z) & k_{21}^j(t, z) & k_{11}^j(t, z) & k_{21}^j(t, z) \end{bmatrix}^T \quad (5.54)$$

with

$$\begin{bmatrix} k_{11}^j(t, z) & k_{12}^j(t, z) \\ k_{21}^j(t, z) & k_{22}^j(t, z) \end{bmatrix} = \begin{bmatrix} -\sin \theta(t, j, z) & -\cos \theta(t, j, z) \\ \cos \theta(t, j, z) & \sin \theta(t, j, z) \end{bmatrix} \begin{bmatrix} k_1 k_t \\ k_2 k_t \end{bmatrix} \begin{bmatrix} -\sin \theta(t, j, z) & \cos \theta(t, j, z) \end{bmatrix} \quad (5.55)$$

Due to the feed motion, the regenerative delay associated with each tooth is time varying in the form

$$\tau_j(t) = \frac{2\pi R}{N[\Omega R + f \cos \theta_j]} \quad (5.56)$$

Matrices  $\hat{\mathbf{K}}(t)$  and  $\hat{\mathbf{K}}_1^j(t, z)$  and the delays  $\tau_j(t)$  are time periodic with the tooth passing period  $T = 2\pi/N\Omega$ , the period of system (5.48) is the tooth pass period  $T$ , and the time step is  $\Delta t = T/K = 2\pi/(KN\Omega)$ . System (5.48) is identical to (5.1), and the same approach discussed previously can be used to determine the stability.

Stability charts are determined with a single degree-of-freedom system for down-milling operations of 25% immersion rate. The corresponding workpiece–tool system modal parameters are given in Table 5.1. The tool and the cutting parameters are provided in Table 5.2. Since, the tool helix angle is zero in this case, the cutting forces along the  $X$ -direction and the  $Y$ -direction do not depend on the normal rake angle and the friction coefficient, both of which are not shown in Table 5.2. The numerical results are compared to experimental results published in [34].

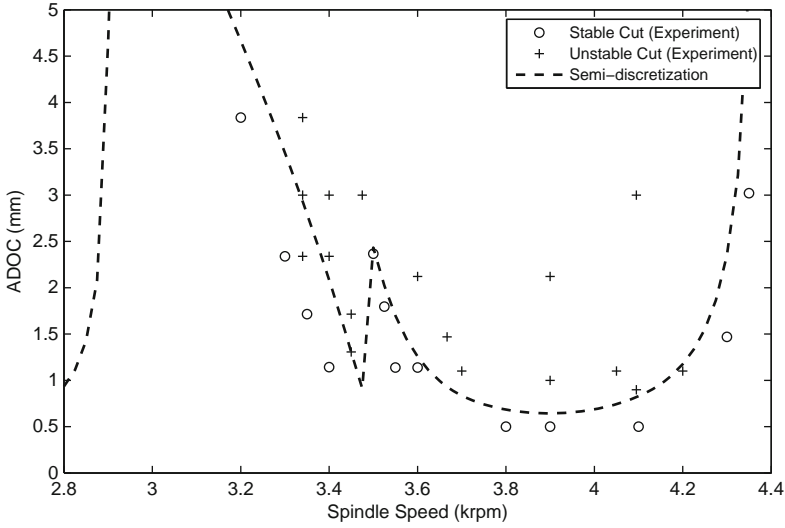
In Fig. 5.9, the stability charts are presented for 25% immersion down-milling operations. Dashed lines denote the stability boundaries obtained by the semidiscretization method. To obtain the stability lobes, the axial depth of cut and the spindle speed are chosen as the control parameters. For a fixed spindle speed, the axial depth of cut is increased gradually in a quasistatic manner. The corresponding pseudomonodromy matrix  $\Phi$  is determined for each pair of the chosen spindle speed and axial depth of cut values. The label “o” (experiment) denotes that the milling process is stable for the chosen experimental control parameters, namely, the axial depth of cut and the spindle speed. The label “+” (experiment) denotes that the milling process is unstable for the corresponding experimental control parameters.

**Table 5.1** Modal parameters of workpiece–tool system

Mode	Frequency (Hz)	Damping (%)	Stiffness (N m <sup>-1</sup> )	Mass (kg)
Workpiece (X)	146.6	0.32.0	$2.18 \times 10^6$	2.573

**Table 5.2** Tool and cutting parameters

Helix angle ( $\eta$ )	Tooth number	Radius (mm)	$K_t$ (MPa)	$k_n$
0°	1	9.53	550	0.364



**Fig. 5.9** Stability predictions for 25% immersion down-milling operations

From this figure, one can find that the stability charts determined by the semidiscretization method agree well with the experimental results.

#### 5.4.5 VSS Milling Process: Nonautonomous DDE with Time-Periodic Delay

VSS milling can be used to suppress chatter that develops during conventional, constant speed machining. During a VSS milling operation, the spindle speed is variable and the time delay is time varying, as well. In this effort, the time variation is a sinusoidal modulation of spindle speed given by

$$\Omega(t) = \Omega_0 + \Omega_1 \sin(\omega_m t) = \Omega_0 [1 + RVA \sin(RVF \cdot \Omega_0 t)], \quad (5.57)$$

where  $\Omega_0$  is the nominal spindle speed,  $\Omega_1$  is amplitude of speed variation,  $\omega_m$  is frequency of speed variation,  $RVA = \Omega_1/\Omega_0$  is the ratio of speed variation amplitude to the nominal spindle speed, and  $RVF = \omega_m/\Omega_0$  is the ratio of the speed variation frequency to the nominal spindle speed. In the case of a VSS milling operation, the angular position  $\theta(t, j, z)$  of tooth  $j$  at axial location  $z$  and time  $t$ , introduced in (5.55), is determined by

$$\theta(t, i, z) = \int_0^t \Omega(s) ds - (i-1) \frac{2\pi}{N} - \frac{\tan \eta}{R} z + \theta_0. \quad (5.58)$$

After substituting (5.57) into (5.58), one can obtain

$$\theta(t, i, z) = \Omega_0 t + \frac{RVA}{RVF} [1 - \cos(\omega_m t)] - (i-1) \frac{2\pi}{N} - \frac{\tan \eta}{R} z + \theta_0. \quad (5.59)$$

The time delay, which has been introduced in (5.47), is determined from

$$\int_{t-\tau(t)}^t \Omega(s) ds = \frac{2\pi}{N} \quad (5.60)$$

By substituting (5.57) into (5.60) and integrating, one can obtain

$$\Omega_0 \tau(t) + \frac{\Omega_1}{\omega_m} \cos(\omega_m(t - \tau(t))) - \frac{\Omega_1}{\omega_m} \cos(\omega_m t) = \frac{2\pi}{N}. \quad (5.61)$$

From (5.61), one cannot get a closed form solution for  $\tau(t)$ . However, for “small” RVA and “small” RVF,  $\tau(t)$  can be approximated as

$$\tau(t) \approx \tau_0 [1 - (1 - RVA \sin(\omega_m t - \phi)) RVA \sin(\omega_m t - \phi)], \quad (5.62)$$

where

$$\tau_0 = \frac{2\pi}{N\Omega_0}. \quad (5.63)$$

Similar to the system presented for the CSS milling process, one can write the system for the VSS milling process in compact form as

$$\dot{\mathbf{Q}}(t) = \mathbf{A}(t)\mathbf{Q}(t) + \mathbf{B}_1(t)\mathbf{Q}(t - \tau(t)) + \mathbf{F}(t), \quad (5.64)$$

where  $\mathbf{A}(t)$  is the coefficient matrix associated with present states and  $\mathbf{B}_1$  is the coefficient matrix associated with delayed states.

As in the system (5.48), the coefficient matrices  $\mathbf{A}(t)$ ,  $\mathbf{B}(t)$ , and  $\mathbf{F}(t)$  are piecewise, periodic functions with the period  $T$ . For the CSS milling process, this period  $T = 2\pi/N\Omega_0$ . For the VSS milling process, this period is equal to the period of the spindle speed modulation:  $T = 1/\omega_m$ . Here, it is assumed that the modulation period is an integer multiple of the nominal tooth passing period  $\tau_0 = 2\pi/N\Omega_0$  (otherwise, the system is quasiperiodic). Referring to Fig. 5.8, the workpiece–tool system modal parameters and the tool and cutting parameters are chosen as shown in Tables 5.3 and 5.4.

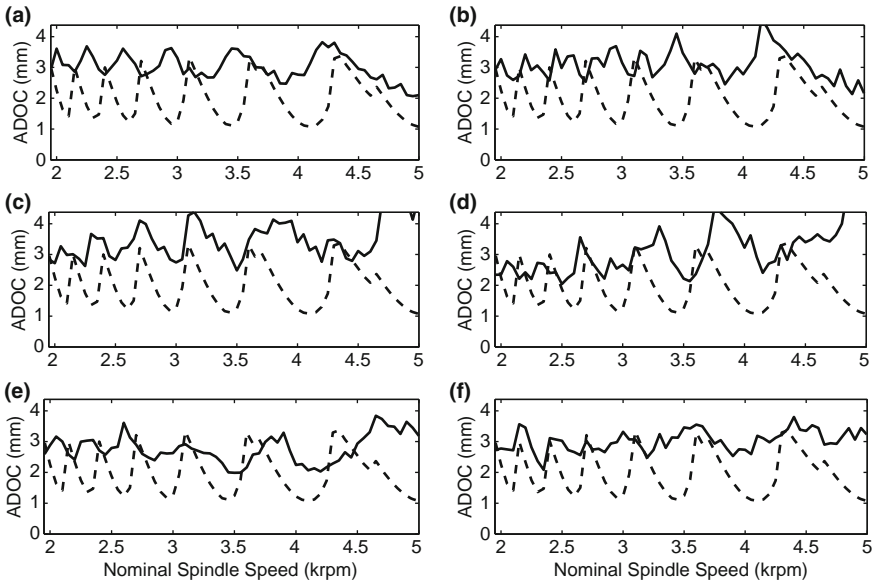
In Fig. 5.10, the stability charts are presented for 5% immersion down-milling operations. Throughout the considered spindle speed range, the range of stable axial depths of cut (ADOC) for VSS milling is larger than that obtained for the

**Table 5.3** Modal parameters of workpiece–tool system

Mode	Frequency (Hz)	Damping (%)	Stiffness (N m <sup>-1</sup> )	Mass (kg)
Tool (X)	729.07	1.07	$9.14 \times 10^5$	$4.36.0 \times 10^{-2}$

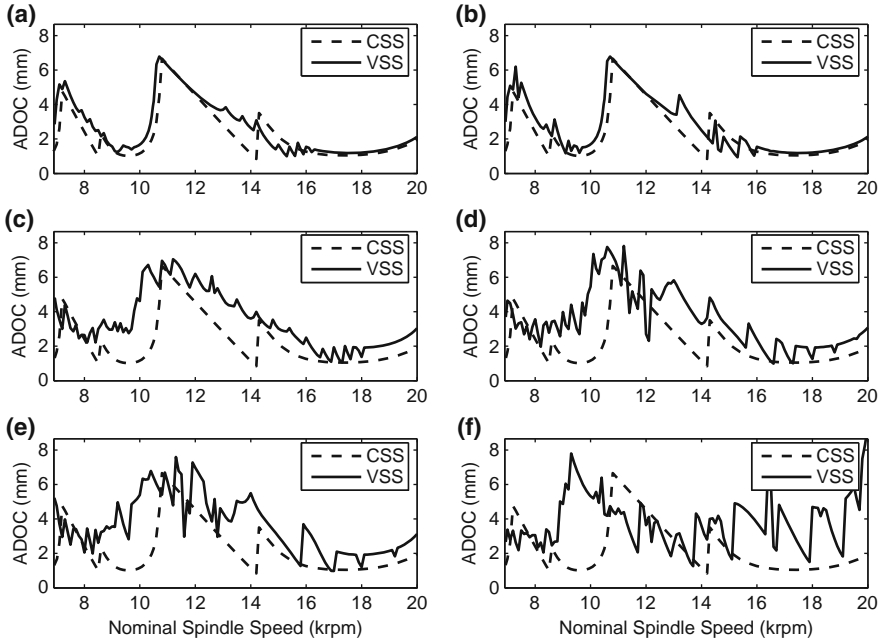
**Table 5.4** Tool and cutting parameters

Normal rake angle ( $\phi_n$ )	Helix angle ( $\eta$ )	Tooth number	Radius (mm)	$K_t$ (MPa)	$k_n$	Cutting friction coefficient ( $\mu$ )
$6^\circ$	$40^\circ$	2	6.35	600	0.42	0.2



**Fig. 5.10** Stability predictions for 5% immersion down-milling operations: (a)  $RVA = 0.1, RVF = 0.1$ , (b)  $RVA = 0.1, RVF = 0.2$ , (c)  $RVA = 0.2, RVF = 0.1$ , (d)  $RVA = 0.2, RVF = 0.2$ , (e)  $RVA = 0.2, RVF = 0.3$ , and (f)  $RVA = 0.3, RVF = 0.3$ ; (—) for CSS and (---) for VSS

corresponding CSS milling operations. The results also show the robustness of the stability of VSS milling process to spindle speed. One can discern the differences among the stability lobes of VSS and CSS operations for spindle speeds up to 5,000 rpm (tooth pass frequency to first natural frequency ratio of 0.23), beyond which the stability lobes are close to each other. For higher spindle speeds, the results are as shown in Fig. 5.11. From this figure, one can infer that the stability lobes for VSS milling are close to those obtained for CSS milling when  $RVA = 0.1$  (Fig. 5.11a, b). For higher values of  $RVA$ ; that is, 0.2 and 0.3, the stability charts are as shown in Fig. 5.11c–f. The stability range is improved and this improvement is also extended to the high-speed range. However, the stable range of ADOC for VSS operations is lower than that obtained for CSS operations in certain spindle speed ranges.



**Fig. 5.11** Stability predictions for 5% immersion down-milling operations: (a)  $RVA = 0.1, RVF = 0.1$ , (b)  $RVA = 0.1, RVF = 0.2$ , (c)  $RVA = 0.2, RVF = 0.1$ , (d)  $RVA = 0.2, RVF = 0.2$ , (e)  $RVA = 0.2, RVF = 0.3$ , and (f)  $RVA = 0.3, RVF = 0.3$ ; (—) for CSS and (---) for VSS

## 5.5 Closure

The first general discussion on Floquet theory for periodic DDEs is due to Stokes [35]. Halanay [8] obtained a general form of the operator  $U_T$  for a linear periodic DDE with constant delay. For DDEs with periodic coefficients and time-varying delays, there is no general form for the operator  $U_T$ . Numerical schemes [9, 10, 36] with an analytical basis are often used to approximate the monodromy matrix  $U_T$ . The semidiscretization method [15] is a numerical method, which can be used to construct the approximated monodromy matrix  $U_T$ , as illustrated in this chapter. This method can be applied to linear DDEs (including piecewise linear DDEs) with periodic coefficients and periodically varying time delays. Since periodic solutions are of interest, the ratio of the period of the coefficients to the period of the time delay needs to be a rational number; that is,  $T_0/T_j$  in (5.1) should be a rational number. The monodromy matrix needs to be considered on the large period for the case of periodic motions, and the obtained information from the analysis can be used to assess local stability. The cost of computations can be high, when eigenvalues of a large monodromy matrix are to be computed. In order to improve the efficiency, Elbeyly and Sun [37] presented the improved zeroth-order and first-order semidiscretization methods, while Insperger et al. [38] have discussed the rate of convergence for zeroth-, first-, and higher order semidiscretization schemes. If the

form of the operator  $U_T$  (or the monodromy operator) for DDEs with time-periodic coefficients and delays can be represented in a general form as in the case of time-periodic DDEs with constant delays [8], one can use this general form to construct an approximate monodromy matrix and study the convergence and efficiency of computations.

**Acknowledgments** X.-H. Long gratefully acknowledges the support received through 973 Grant No. 2005CB724101. T. Insperger was supported by the HNSF under grant no. OTKA K72911 and the János Bolyai Research Scholarship of the HAS.

## References

1. Hale, J.K., History of delay equations, In: Arino O., M.L. Hbid, and Ait D.E. (eds) *Delay differential equations and applications*. Springer, Berlin 2006, 1–28.
2. Bellman, R. and Cooke, K.L., *Differential-differences equations*, Academic, New York, NY, 1963.
3. Stépán, G., *Retarded dynamical systems*, Longman: Harlow, UK, 1989.
4. Altintas, Y. and Budak, E., Alytical prediction of stability lobes in milling, *Annals of CIRP* 44, 1995, 357–362.
5. Zhao, M.X. and Balachandran, B., Dynamics and stability of milling process, *International Journal of Solids and Structures* 38, 2001, 2233–2248.
6. Hale, J.K. and Lunel, S. M., *Introduction to functional differential equations*, Springer, New York, NY, 1993.
7. Hahn, W., On difference-differential equations with periodic coefficients, *Journal of Mathematical Analysis and Applications* 3, 1961, 70–101.
8. Halanay, S., *Differential equations: stability, oscillations, time lags*, Academic, New York, NY, 1966.
9. Butcher, E.A., Ma, H.T., Bueler, E., Averina, V., and Szabo, Z., Stability of linear time-periodic delay-differential equations via Chebyshev polynomials, *International Journal for Numerical Methods in Engineering* 59, 2004, 895–922.
10. Bayly, P.V., Halley, J.E., Mann, B.P., and Davis, M.A., Stability of interrupted cutting by temporal finite element analysis, *Journal of Manufacturing Science and Engineering* 125, 2003, 220–225.
11. Insperger, T., Mann, B.P., Stépán, G., and Bayly, P.V., Stability of up-milling and down-milling, part I: alternative analytical methods, *International Journal of Machine Tools and Manufacture* 43, 2003, 25–34.
12. Mann, B.P., Bayly, P.V., Davies, M.A., and Halley, J.E., Limit cycles, bifurcations, and accuracy of the milling process, *Journal of Sound and Vibration*, 277, 2004, 31–48.
13. Yilmaz, A., Al-Regib, E., and Ni, J., Machine tool chatter suppression by multi-level random spindle speed variation, *ASME Journal of Manufacturing Science and Engineering* 124, 2002, 208–216.
14. Minis, I. and Yanushevsky, R., A new theoretical approach for the prediction of machine tool chatter in milling, *ASME Journal of Engineering for Industry* 115, 1993, 1–8.
15. Insperger, T. and Stépán, G., Semi-discretization method for delayed systems, *International Journal of Numerical Methods in Engineering* 55, 2002, 503–518.
16. Insperger, T. and Stépán, G., Updated semi-discretization method for periodic delay-differential equations with discrete delay, *International Journal of Numerical Methods in Engineering* 61(1), 2004, 117–141.
17. Long, X.H. and Balachandran, B. Stability of milling process, *Nonlinear Dynamics* 49, 2007, 349–359.

18. Long, X.H., Balachandran, B., and Mann, B.P. Dynamics of milling processes with variable time delays, *Nonlinear Dynamics* 47, 2007, 49–63.
19. Liu, Z. and Liao, L., Existence and global exponential stability of periodic solution of cellular neural networks with time-varying delays, *Journal of Mathematical Analysis and Applications* 290, 2004, 247–262.
20. Ruan, S., Delay Differential equations in single species dynamics, In: Arino O, Hbid M.L., and Ait D.E. (eds) *Delay differential equations and applications*. Springer, Berlin Heidelberg New York, 2006, 477–518.
21. Schely, D. and Gourley, S.A., Linear stability criteria for population models with periodic perturbed delays, *Journal of Mathematical Biology* 40, 2000, 500–524.
22. Jiang, M.H., Shen, Y., and Liao, X.X., Global stability of periodic solution for bidirectional associative memory neural networks with varying-time delay, *Applied Mathematics and Computation* 182, 2006, 509–520.
23. Zhou, Q.H., Sun, J.H., and Chen, G.R., Global exponential stability and periodic oscillations of reaction-diffusion BAM neural networks with periodic coefficients and general delays, *International Journal of Bifurcation and Chaos* 17, 2007, 129–142.
24. Altintas, Y. and Chan, P.K., In-process detection and suppression of chatter in milling, *International Journal of Machine Tools and Manufacture* 32, 1992, 329–347.
25. Radulescu, R., Kapoor, S.G., and DeVor, R.E., An investigation of variable spindle speed face milling for tool-work structures with complex dynamics, part 1: simulation results, *ASME Journal of Manufacturing Science and Engineering* 119, 1997, 266–272.
26. Radulescu, R., Kapoor, S.G., and DeVor, R.E., An investigation of variable spindle speed face milling for tool-work structures with complex dynamics, part 2: physical explanation, *ASME Journal of Manufacturing Science and Engineering* 119, 1997, 273–280.
27. Tsao, T.C., McCarthy, M.W., and Kapoor, S.G., A new approach to stability analysis of variable speed machining systems, *International Journal of Machine Tools and Manufacture* 33, 1993, 791–808.
28. Sastry, S., Kapoor, S.G., DeVor, R.E., and Dullerud, G.E., Chatter stability analysis of the variable speed face-milling process, *ASME Journal of Manufacturing Science and Engineering* 123, 2001, 753–756.
29. Sastry, S., Kapoor, S.G., and DeVor, R.E., Floquet theory based approach for stability analysis of the variable speed face-milling process, *ASME Journal of Manufacturing Science and Engineering* 124, 2002, 10–17.
30. Long, X.H. and Balachandran, B., Stability of up-milling and down-milling operations with variable spindle speed, *Journal of Vibration and Control*, 2008, accepted for publication.
31. Krasovskii, N.N., *Stability of motion*, Stanford University Press, Palo Alto, CA, 1963.
32. Nayfeh, A.H. and Balachandran, B., *Applied nonlinear dynamics: Analytical, computational, and experimental methods*, Wiley, New York, NY, 1995.
33. Insperger, T. and Stépán, G., Stability of the damped Mathieu equation with time delay, *Journal of Dynamics System, Measurement and Control*, 125, 2003, 166–171.
34. Mann, B.P., Insperger, T., Bayly, P.V., and Stépán, G., Stability of up-milling and down-milling, Part 2: Experimental verification, *International Journal of Machine Tools and Manufacture* 43, 2003, 35–40.
35. Stokes, A.P., A Floquet theory for functional differential equations, *The Proceedings of the National Academy of Sciences U.S.A.* 48, 1962, 1330–1334.
36. Gilsinn, D.E., and Potra, F.A. Integral Operators and Delay Differential Equations, *Journal of Integral Equations and Applications* 18, 2006, 297–336.
37. Elbeyly, O. and Sun, J.Q., On the semi-discretization method for feedback control design of linear systems with time delay, *Journal of Sound and Vibration* 273, 2004, 429–440.
38. Insperger, T., Stépán, G., and Turi, J., On the higher-order semi-discretizations for periodic delayed systems, *Journal of Sound and Vibration* 313, 2008, 334–341.

# Chapter 6

## Bifurcations, Center Manifolds, and Periodic Solutions

David E. Gilsinn

**Abstract** Nonlinear time-delay differential equations are well known to have arisen in models in physiology, biology, and population dynamics. These delay differential equations (DDEs) usually have parameters in their formulation. How the nature of the solutions change as the parameters vary is crucial to understanding the underlying physical processes. When the DDE is reduced, at an equilibrium point, to leading linear terms and the remaining nonlinear terms, the eigenvalues of the leading coefficients indicate the nature of the solutions in the neighborhood of the equilibrium point. If there are any eigenvalues with zero real parts, periodic solutions can arise. One way in which this can happen is through a bifurcation process called a Hopf bifurcation in which a parameter passes through a critical value and the solutions change from equilibrium solutions to periodic solutions. This chapter describes a method of decomposing the DDE into a form that isolates the study of the periodic solutions arising from a Hopf bifurcation to the study of a reduced size differential equation on a surface, called a center manifold. The method will be illustrated by Hopf bifurcation that arises in machine tool dynamics, which leads to a machining instability called regenerative chatter.

**Keywords:** Center manifolds · Delay differential equations · Exponential polynomials · Hopf bifurcation · Limit cycle · Machine tool chatter · Normal form Semigroup of operators · Subcritical bifurcation

### 6.1 Background

With the advent of new technologies for measurement instrumentation, it has become possible to detect time delays in feedback signals that affect a physical

---

Contribution of the National Institute of Standards and Technology, a Federal Agency. Not subject to copyright.

system's performance. System models in a number of fields have been investigated in which time delays have been introduced in order to have the output of the models more closely reflect the measured performance. These fields have included physiology, biology, and population dynamics (see an der Heiden [2], Kuang [23], and MacDonald [24]).

In recent years, time delays have arisen in models of machine tool dynamics. In particular, a phenomenon, called regenerative chatter, is being heavily studied. Regenerative chatter can be recognized on a manufacturing plant floor by a characteristic high-pitched squeeling sound, distinctive marks on the workpiece, and by undulated or dissected chips (see Tlustý [32]). It is a self-excited oscillation of the cutting tool relative to the workpiece during machining. Self-excited oscillations, mathematically called limit cycles or isolated periodic solutions, reflect the fact that there are nonlinearities in the physical system being modeled that have to be taken into account. For further reading on delay differential equations (DDEs) in turning or numerically controlled lathe operations the reader is referred to Kalmár-Nagy et al. [20] and [21]. For problems in drilling see Stone and Askari [29] and Stone and Campbell [30]. Finally, for problems in milling operations see Balachandran [5] and Balachandran and Zhao [6]. The modeling of regenerative chatter arose in work at the National Institute of Standards and Technology (NIST) in conjunction with work related to error control and measurement for numeric control machining.

Along with the nonlinearities, the differential equations, from ordinary, partial, and delay equations, that model the physical processes, usually depend on parameters that have physical significance, such as mass, fundamental system frequency, nonlinear gains, and levels of external excitation. Changes, even small ones, in many of these system parameters can drastically change the qualitative nature of the system model solutions. In this chapter, we will examine the effect that variations of these parameters have on the nature and the number of solutions to a class of nonlinear DDEs. The changing nature of solutions to a differential equation is often referred to as a bifurcation, although formally the concept of bifurcation refers to parameter space analysis. The term bifurcation means a qualitative change in the number and types of solutions of a system depending on the variation of one or more parameters on which the system depends. In this chapter, we will be concerned with bifurcations in the nature of solutions to a DDE that occur at certain points in the space of parameters, called Hopf bifurcation points. The bifurcations that arise will be called Hopf bifurcations.

From an assumed earlier course in differential equations, it should be clear to the reader that the eigenvalues of the linear portion of the state equations are an indicator of the nature of the solutions. For example, if all of the eigenvalues have negative real parts then we can expect the solutions to be stable in some sense and if any of them have positive real parts then we can expect some instabilities in the system. What happens if any of the eigenvalues have zero real parts? This is where, one might say, the mathematical fun begins, because these eigenvalues indicate that there is likely to be some oscillatory affects showing up in the solutions. The game then is to first determine those system parameters that lead to eigenvalues with zero real parts. The next step in analyzing a system of differential equations, that depends on

parameters, is to write the system in terms of its linear part and the remaining non-linear part and then to decompose it in order to isolate those equation components most directly affected by the eigenvalues with zero real parts and those equations affected by the eigenvalues with nonzero real parts. This same approach applies to problems both in ordinary differential equations as well as in DDEs. Once this decomposition has been developed we can then concentrate our effort on studying the component equations related to the eigenvalues with zero real parts and apply methods to simplify them.

We will assume at this point that a time-delay differential equation modeling a physical phenomenon has been written with linear and nonlinear terms. Although DDEs come in many forms, in this chapter we will only consider delay equations of the form

$$\frac{dz}{dt}(t, \mu) = U(\mu)z(t, \mu) + V(\mu)z(t - \sigma, \mu) + f(z(t, \mu), z(t - \sigma), \mu), \quad (6.1)$$

where  $z \in R^n$ , the space of  $n$ -dimensional real numbers;  $U$  and  $V$ , the coefficient matrices of the linear terms, are  $n \times n$  matrices;  $f \in R^n$ ,  $f(0, 0, \mu) = 0$ , is a nonlinear function;  $\mu$  is a system parameter; and  $\sigma, s \in R$ . For most practical problems we can assume the  $f$  function in (6.1) is sufficiently differentiable with respect to the first and second variables and with respect to the parameter  $\mu$ . Let  $\mathcal{C}_0$  be the class of continuous functions on  $[-\sigma, 0]$  and let  $z(0) = z_0 \in R^n$ . Then, there exists, at least locally, a unique solution to (6.1) that is not only continuous but is also differentiable with respect to  $\mu$ . For a full discussion of the existence, uniqueness, and continuity questions for DDEs the reader is referred to Hale and Lunel [15]. For ordinary differential equations see the comparable results in Cronin [10]. For the rest of this chapter we will assume that, for both ordinary and delay differential equations, unique solutions exist and that they are continuous with respect to parameters.

Although the results discussed in this chapter can be extended to higher dimension spaces, we will mainly be interested in problems with  $n = 2$  that are dependent on a single parameter. We will also concentrate on bifurcations in the neighborhood of the  $z(t) \equiv 0$  solution. This is clearly an equilibrium point of (6.1) and is referred to as a local bifurcation point. For a discussion of various classes of bifurcations see Nayfeh and Balachandran [26]. Since machine tool chatter occurs when self-oscillating solutions emanate from equilibrium points, i.e., stable cutting, we will concentrate in this chapter on a class of bifurcations called Hopf (more properly referenced as Poincaré–Andronov–Hopf in Wiggins [34]) bifurcations. These are bifurcations in which a family of isolated periodic solutions arises as the system parameters change and the eigenvalues cross the imaginary axis. As earlier noted, the equilibrium points at which Hopf bifurcations occur are sometimes referred to as Hopf points. The occurrence of Hopf bifurcations depends on the eigenvalues of the linear portion of (6.1), given by

$$\frac{dz}{dt}(t, \mu) = U(\mu)z(t, \mu) + V(\mu)z(t - \sigma, \mu), \quad (6.2)$$

in which at least one of the eigenvalues of this problem has a zero real part.

As in ordinary differential equations, the eigenvalues of the linear system tell the nature of the stability of solutions of both (6.1) and (6.2). In ordinary differential equations the eigenvalues are computed from the characteristic polynomial and in DDEs the eigenvalues arise from an equation called the *characteristic equation* associated with the linear equation (6.2). This equation is a transcendental equation with an infinite number of solutions called the eigenvalues of (6.2). We will assume that  $z(t) \equiv 0$  is an equilibrium point of (6.1) and that, in the neighborhood of this equilibrium point, (6.2) has a family of pairs of eigenvalues,  $\lambda(\mu)$ ,  $\bar{\lambda}(\mu)$ , of (6.2) such that  $\lambda(\mu) = \alpha(\mu) + i\omega(\mu)$ , where  $\alpha$ ,  $\omega$  are real,  $\alpha(0) = 0$ ,  $\omega(\mu) > 0$ ,  $\alpha'(0) \neq 0$ . This last condition is called a transversality condition and implies that the family of eigenvalues  $\lambda(\mu) = \alpha(\mu) + i\omega(\mu)$  is passing across the imaginary axis. Under these conditions a Hopf bifurcation occurs, where periodic solutions arise from the equilibrium point. These conditions also apply in the case of ordinary differential equations.

In both ordinary and delay differential equations, the nature of the stability of these bifurcating periodic solutions at Hopf points can be more easily studied if (6.1), in the case of DDEs, can be reduced to a simpler form in the vicinity of the bifurcation point. We will develop a simplification technique comparable to that used in ordinary differential equations. It reduces the study of the stability of the bifurcating periodic solutions to the study of periodic solutions of a simplified system, called a normal form, on a surface, called a center manifold. Center manifolds arise when the real parts of some eigenvalues are zero. A center manifold is an invariant manifold in that, if solutions to (6.1) begin on the manifold, they remain on the manifold. This manifold is usually of lower dimension than the space and the nature of the stability of the equilibrium point depends on the projected form of (6.1) on the manifold. The projection of (6.1) onto the center manifold usually leads to a lower order system and the conversion of that system to a normal form provides a means of studying the stability of the bifurcating periodic solutions. In fact we will see that, for the example machining problem, the analysis will reduce to a problem of solving a simplified approximate ordinary differential equation on a center manifold of dimension two. The reduction process is carried out in a number of steps based on the approaches of Hassard et al. [17] and Wiggins [34].

The bifurcation analysis developed in this chapter depends on three simplification steps. In the first step, we will show how the DDE (6.1) can be transformed into three equations, where the eigenvalues of the first two have zero real parts and the eigenvalues of the third have negative real parts. In the process of developing these equations, we will show how the transformations involved are analogous to those used in ordinary differential equations. In the second simplification step, we will show the form that (6.1) takes on the center manifold and finally, in the third step, we will see what the normal form looks like and, from this, how the bifurcating periodic solutions are developed. We will develop the first decomposition more thoroughly because it exemplifies the analogies between ordinary and delay differential equations. The other simplifications will be given as formulas, but an example will be given at the end in which the details of the transformations in a particular case are developed and used to predict stability of the bifurcating solutions. For a more

automated approach to computing a center manifold, using a symbolic manipulation program, the reader is referred to Campbell [8] as well as to the eighth chapter of this book.

This is a note to the reader. In this sixth chapter, we will be dealing with differential equations, both ordinary and delay, that have complex eigenvalues and eigenvectors. Although the differential equations associated with the real-world models are most often formulated in the real space  $R^n$ , these differential equations can be viewed as being embedded in the complex space  $C^n$ . This process is called *complexification* and allows the differential equation to be studied or solved in the complex space. At the end, appropriate parts of the complex solution are identified with solutions to the original differential equation in the real domain. A thorough discussion of this process is given in Hirsch and Smale [19]. We will not be concerned with the formal embedding of the real differential equations in the complex domain, but just to note that it is possible and causes no difficulties with what we will be talking about in this chapter. We only bring this up so that the reader may understand why we seem to flip back and forth between real and complex equations. We are simply working with this complexification process behind the scenes without writing out the complete details. As we work through the examples, the reader will see that it is quite natural to exist in both domains and see that the final result leads to the desired approximate solution of the original DDE.

The chapter is divided as follows. In Sect. 6.2, we will show how the adjoint to a linear ordinary differential equation can be used to naturally generate a bilinear form that acts as an inner product substitute and introduces geometry to a function space. This bilinear form is then used to define an orthogonality property that is used to decompose the differential equations into those equations that have eigenvalues with zero real parts and those that have nonzero real parts. We first introduce these ideas for ordinary differential equations in order to show that the ideas are models for the analogous ideas for DDEs. In Sect. 6.3, we will show how this decomposition works for an ordinary differential equation example. In Sect. 6.4, we will show how a DDE can be formulated as an operator equation that has a form similar to an ordinary differential equation. Within that section we will also introduce a bilinear form by way of an adjoint equation that is analogous to the one in ordinary differential equation and is used in a similar manner to decompose the DDE into a system of operator equations that have components dependent on the eigenvalues of the characteristic equation with zero real parts and those with nonzero real parts. In Sect. 6.5, the author starts by introducing the main example of a DDE, which we will consider in this chapter. We will show how it is reduced to an operator equation and decomposed into components dependent on eigenvalues with zero real parts and those dependent on non-zero real parts. In Sect. 6.6, we will introduce the general formulas needed in order to compute the center manifold, the normal form of the DDE on the center manifold, and the bifurcated periodic solution for the DDE on the manifold. In Sect. 6.7 we will continue with the main example and develop the form of the center manifold, the normal form for the example on the center manifold, and finally the resulting periodic solution on the center manifold. In Sect. 6.8, we show the results of numerical simulations of the example DDE in the vicinity

of the bifurcation points and show that there is a possibility of unstable behavior for values of system parameters that extend into parameter regions, which would otherwise be considered stable.

## 6.2 Decomposing Ordinary Differential Equations Using Adjoints

Many results for DDEs are direct analogies of results in ordinary differential equations. In this section, we will review some properties of ordinary differential equations that motivate analogous properties in DDEs. We will break the decomposition process down into five basic steps. Later we will show that analogous five steps can be used to decompose a DDE.

### 6.2.1 Step 1: Form the Vector Equation

We begin by considering the differential equation

$$\frac{dz}{dt}(t) = Az(t) + f(z(t), \mu), \quad (6.3)$$

$z \in \mathbb{R}^n$ ,  $A$  an  $n \times n$  real matrix,  $f \in \mathbb{R}^n$  with locally bounded derivatives,  $-\infty < t < \infty$ . The homogeneous part is given by

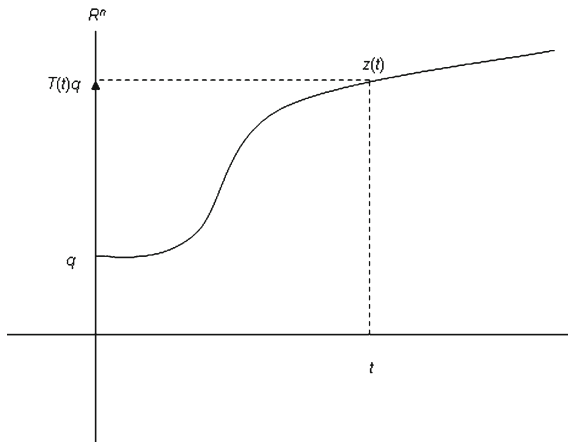
$$\frac{dz}{dt}(t) = Az(t). \quad (6.4)$$

We will stay as much as we can with real variables since many of the problems leading to DDEs are formulated with real variables.

The solution of the linear system (6.4) with constant coefficients can be represented as a parametric operator of the form

$$T(t)q = z_t(q) = e^{At}q, \quad (6.5)$$

acting on a vector  $q \in \mathbb{R}^n$ .  $T(t)$  is said to be a *group* of operators since  $T(t_1 + t_2) = T(t_1)T(t_2)$  and  $T(t)T(-t) = I$ , the identity. The family of operators would be called a *semigroup* if an identity exists but there are no inverse elements. In Fig. 6.1 we show two points of view about solutions to differential equations. Traditionally we look at solutions to ordinary differential equations as trajectories beginning with an initial vector  $q$  and taking on the value  $z(t)$  at some time  $t > 0$ , say. There is a sense of physical meaning here such as the trajectory of a ball. However, the solutions can also be viewed as  $t$ -dependent maps  $T(t)$  in  $\mathbb{R}^n$  of initial vectors  $q$ . They can then be thought of as a flow in  $\mathbb{R}^n$  space. This is a point of view taken by Arnold [3]. It will be this point of view that will turn out more fruitful when we get to DDEs.



**Fig. 6.1** The solution operator maps functions in  $R^n$  to functions in  $R^n$

In most ordinary differential equation textbooks, a linear transformation is used to write the matrix  $A$  in a form that separates it into blocks with eigenvalues having positive, zero, or negative real parts, called a Jordan normal form. However, in this section, we will introduce a change of coordinates procedure that accomplishes the decomposition by way of the adjoint equation since this translates to an analogous method for DDEs. This process allows us to introduce a geometric point of view to the decomposition of (6.3). In particular, we can use the geometric property of orthogonality as a tool to decompose (6.3).

### 6.2.2 Step 2: Define the Adjoint Equation

The adjoint equation to (6.4) is given by

$$\frac{dy}{dt}(t) = -A^T y(t). \quad (6.6)$$

### 6.2.3 Step 3: Define a Natural Inner Product by way of an Adjoint

Equations (6.4) and (6.6) are related by the Lagrange identity

$$y^T \Theta z + \Omega y^T z = \frac{d}{dt} (y^T z), \quad (6.7)$$

where  $\Theta z = \dot{z} - Az$ ,  $\Omega y = \dot{y} + A^T y$ . If  $z$  and  $y$  are solutions of (6.4) and (6.6), respectively, then it is clear that  $(d/dt)(y^T z) = 0$  which implies  $y^T z$  is constant and is the

natural inner product of  $R^n$ . We note here that in case if the inner product is taken, where  $y$  and  $z$  are complex, then the inner product would be  $\bar{y}^T z$ . Thus, the use of an adjoint equation leads naturally to an inner product definition. It might then seem reasonable that using adjoint properties could geometrically lead to some form of *orthogonal* decomposition of a system of differential equations in a similar manner to the process of orthogonal decomposition of a vector in  $R^n$ . In fact this is what we will show in this section and in Sect. 6.4 we will show that the same idea extends to DDEs.

We begin by stating some results on eigenvalues and eigenvectors from linear algebra that have direct analogs in the delay case. To be general, we will state them in the complex case. Let  $A$  be an  $n \times n$  matrix with elements in  $C^n$  and  $A^*$  the usual conjugate transpose matrix. Let  $(\cdot, \cdot)$  be the ordinary inner product in  $C^n$ . Then the following hold.

1.  $(\bar{\psi}, A\phi) = (A^*\bar{\psi}, \phi)$ .
2.  $\lambda$  is an eigenvalue of  $A$  if and only if  $\bar{\lambda}$  is an eigenvalue of  $A^*$ .
3. The dimensions of the eigenspaces of  $A$  and  $A^*$  are equal.
4. Let  $\phi_1, \dots, \phi_d$  be a basis for the right eigenspace of  $A$  associated with eigenvalues  $\lambda_1, \dots, \lambda_d$  and let  $\psi_1, \dots, \psi_d$  be a basis for the right eigenspace of  $A^*$  associated with the eigenvalues  $\bar{\lambda}_1, \dots, \bar{\lambda}_d$ . Construct the matrices  $\Phi = (\phi_1, \dots, \phi_d)$ ,  $\Psi = (\psi_1, \dots, \psi_d)$ . The matrices  $\Phi$  and  $\Psi$  are  $n \times d$ . If we define the bilinear form

$$\langle \Psi, \Phi \rangle = \begin{pmatrix} (\psi_1, \phi_1) & \cdots & (\psi_1, \phi_d) \\ \vdots & \ddots & \vdots \\ (\psi_d, \phi_1) & \cdots & (\psi_d, \phi_d) \end{pmatrix}, \quad (6.8)$$

then  $\langle \Psi, \Phi \rangle$  is nonsingular and can be chosen so that  $\langle \Psi, \Phi \rangle = I$ .

Although the  $\langle \cdot, \cdot \rangle$  is defined in terms of  $\Psi$  and  $\Phi$ , the definition is general and can be applied to any two matrices  $U$  and  $Z$ . Thus,  $\langle U, Z \rangle$ , where  $U$  and  $Z$  are matrices, is a bilinear form that satisfies properties of an inner product. In particular

$$\begin{aligned} \langle U, \alpha Z_1 + \beta Z_2 \rangle &= \alpha \langle U, Z_1 \rangle + \beta \langle U, Z_2 \rangle, \\ \langle \alpha U_1 + \beta U_2, Z \rangle &= \bar{\alpha} \langle U_1, Z \rangle + \bar{\beta} \langle U_2, Z \rangle, \\ \langle UM, Z \rangle &= M^* \langle U, Z \rangle, \\ \langle U, ZM \rangle &= \langle U, Z \rangle M, \end{aligned} \quad (6.9)$$

where  $\alpha, \beta$  are complex constants and  $M$  is a compatible matrix.

#### 6.2.4 Step 4: Get the Critical Eigenvalues

Although  $A$  in (6.3) is real, it can have multiple real and complex eigenvalues. The complex ones will appear in pairs. To reduce computational complexity, we will assume that  $A$  has two eigenvalues  $i\omega$ ,  $-i\omega$  with associated eigenvectors  $\phi$ ,  $\bar{\phi}$ . We

will assume that all other eigenvalues are distinct from these and have negative real parts. The associated eigenvalues with zero real parts of  $A^*$  are  $-i\omega$ ,  $i\omega$  with right eigenvectors  $\bar{\psi}$ ,  $\psi$ . We will use properties of the adjoint to decompose (6.3) into three equations in which the first two will have eigenvalues with zero real parts.

### 6.2.5 Step 5: Apply Orthogonal Decomposition

We begin by defining the matrices

$$\Phi = (\phi, \bar{\phi}), \quad \Psi = (\bar{\psi}, \psi)^T. \quad (6.10)$$

Here we take  $d = 2$  in (6.8) and note that  $\Phi$  and  $\Psi$  are  $n \times 2$  matrices.

Let  $z(t, \mu)$  be the unique family of solutions of (6.3). This is possible due to the standard existence, uniqueness, and continuity theorems. Define

$$Y(t, \mu) = \langle \Psi, z(t, \mu) \rangle = \begin{pmatrix} (\bar{\psi}, z(t, \mu)) \\ (\psi, z(t, \mu)) \end{pmatrix}, \quad (6.11)$$

where  $Y(t, \mu) \in \mathbb{C}^2$ . Set  $Y(t, \mu) = (y_1(t, \mu), y_2(t, \mu))^T$ , where  $y_1(t, \mu) = (\bar{\psi}, z(t, \mu))$  and  $y_2(t, \mu) = (\psi, z(t, \mu))$ .

Define the matrices

$$B = \begin{pmatrix} i\omega & 0 \\ 0 & -i\omega \end{pmatrix}, \quad B^* = \begin{pmatrix} -i\omega & 0 \\ 0 & i\omega \end{pmatrix}. \quad (6.12)$$

They satisfy  $A\Phi = \Phi B$ ,  $A^*\Psi = \Psi B^*$ . If we join (6.9), (6.11), and (6.12) with the fact that  $B^{**} = B$  then we have

$$\frac{dY}{dt}(t, \mu) = BY(t, \mu) + \langle \Psi, f(z(t, \mu), \mu) \rangle. \quad (6.13)$$

This can be written as

$$\begin{aligned} \frac{dy}{dt}(t, \mu) &= i\omega y(t, \mu) + F(t, \mu), \\ \frac{d\bar{y}}{dt}(t, \mu) &= -i\omega \bar{y}(t, \mu) + \bar{F}(t, \mu), \end{aligned} \quad (6.14)$$

where  $F(t, \mu) = (\bar{\psi}, f(z(t, \mu), \mu))$ . These are the first two of the three equations.

In order to develop the third equation, we will use a notion involved with decomposing a vector into two orthogonal components. Here is where geometry enters the picture. This decomposition process arises, for example, when one applies the Gramm–Schmidt orthogonalization method numerically to vectors. The general idea is that the difference between a vector and its orthogonal projection on a linear space, formed from previous orthogonalized vectors, generates an orthogonal decomposition of the original vector. This idea will be generalized here to function spaces, but the basic methodology holds true.

Begin by defining the difference between  $z(t, \mu)$  and its projection onto the linear space formed by the columns of  $\Phi$  as

$$w(t, \mu) = z(t, \mu) - \Phi Y(t, \mu), \quad (6.15)$$

which makes sense in terms of dimensionality, since  $\Phi$  is an  $n \times 2$  matrix,  $Y(t, \mu) \in C^2$ , and  $z \in R^n$ . This is a vector orthogonal to  $\Phi Y(t, \mu)$  in function space. We note that  $w(t, \mu)$  is real because  $\Phi Y(t, \mu) = (\bar{\Psi}, z(t, \mu))\phi + (\Psi, z(t, \mu))\bar{\phi} = 2\text{Re}\{(\bar{\Psi}, z(t, \mu))\phi\}$ .

To show that this function is orthogonal to the space formed by the columns of  $\Phi$  we note that

$$\begin{aligned} w(t, \mu) &= z(t, \mu) - \Phi \langle \Psi, z(t, \mu) \rangle, \\ \langle \Psi, w(t, \mu) \rangle &= \langle \Psi, z(t, \mu) \rangle - \langle \Psi, \Phi \langle \Psi, z(t, \mu) \rangle \rangle, \\ &= \langle \Psi, z(t, \mu) \rangle - \langle \Psi, \Phi \rangle \langle \Psi, z(t, \mu) \rangle, \\ &= \langle \Psi, z(t, \mu) \rangle - \langle \Psi, z(t, \mu) \rangle = 0, \end{aligned} \quad (6.16)$$

where we have used a property from (6.9) and the fact that  $\langle \Psi, \Phi \rangle = I$ .

Now let  $f(z, \mu) = f(z(t, \mu), \mu)$ , and use (6.3), (6.11), and (6.13) to show

$$\begin{aligned} \frac{dw}{dt}(t, \mu) &= \frac{dz}{dt}(t, \mu) - \Phi \frac{dY}{dt}(t, \mu) \\ &= Az(t, \mu) + f(z, \mu) - \Phi BY(t, \mu) - \Phi \langle \Psi, f(z, \mu) \rangle, \end{aligned} \quad (6.17)$$

Substitute  $z(t, \mu) = w(t, \mu) + \Phi Y(t, \mu)$  into (6.17) and use  $A\Phi = \Phi B$  to get

$$\frac{dw}{dt}(t, \mu) = Aw(t, \mu) + f(z, \mu) - \Phi \langle \Psi, f(z, \mu) \rangle. \quad (6.18)$$

Therefore we have the final decomposition as

$$\begin{aligned} \frac{dy}{dt}(t, \mu) &= i\omega y(t, \mu) + F(t, \mu), \\ \frac{d\bar{y}}{dt}(t, \mu) &= -i\omega \bar{y}(t, \mu) + \bar{F}(t, \mu), \\ \frac{dw}{dt}(t, \mu) &= Aw(t, \mu) - \Phi \langle \Psi, f(z, \mu) \rangle + f(z, \mu). \end{aligned} \quad (6.19)$$

After defining some operators in Sect. 6.4, that will take the place of the matrices used here, we will see that there is an analogous decomposition for DDEs.

### 6.3 An Example Application in Ordinary Differential Equations

In this example, we will consider a simple ordinary differential equation and work through the details of the decomposition described in this section. In Sect. 6.4, we will begin working out a more extensive example in DDEs and show how the

decomposition in the time-delay case is analogous to the decomposition in the ordinary differential equations case. In this section, we will follow the decomposition steps give in Sect. 6.2.

Start with the equation

$$\ddot{x} + x = \mu x^2. \quad (6.20)$$

From the existence and uniqueness theorem we know there exists a unique solution  $x(t, \mu)$ , given the initial conditions  $x(0, \mu) = x_0$ ,  $\dot{x}(0, \mu) = x_1$ , that is continuous with respect to  $\mu$ .

### 6.3.1 Step 1: Form the Vector Equation

If we let  $z_1 = x, z_2 = \dot{x}$  then (6.20) can be written in vector form as

$$\dot{z}(t, \mu) = Az(t, \mu) + f(z(t, \mu), \mu), \quad (6.21)$$

where  $z(t, \mu) = (z_1(t, \mu), z_2(t, \mu))^T$ ,  $f(z(t, \mu), \mu) = \mu (0, z_1(t, \mu)^2)^T$ , and

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \quad (6.22)$$

The linear part of (6.21) is

$$\dot{z}(t, \mu) = Az(t, \mu). \quad (6.23)$$

### 6.3.2 Step 2: Define the Adjoint Equation

The adjoint equation of (6.23) is given by

$$\dot{y}(t, \mu) = -A^T z(t, \mu). \quad (6.24)$$

### 6.3.3 Step 3: Define a Natural Inner Product by Way of an Adjoint

The inner product is developed, as in Sect. 6.2, as the natural inner product of vectors. We will go directly to forming the basis vectors.

A basis for the right eigenspace of  $A$  can easily be computed as

$$\phi = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} \end{pmatrix}, \quad \bar{\phi} = \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} \end{pmatrix}, \quad (6.25)$$

These are associated, respectively, with the eigenvalues  $\lambda = i$  and  $\lambda = -i$ . The related eigenvalues and eigenvectors for  $A^T$  are  $\lambda = -i$  and  $\lambda = i$  with the respective eigenvectors  $\bar{\phi}$ ,  $\phi$ . The factor  $1/\sqrt{2}$  is a normalization factor. Now define

$$\Phi = (\phi, \bar{\phi}) = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} & -\frac{i}{\sqrt{2}} \end{pmatrix}, \quad (6.26)$$

and

$$\Psi = (\bar{\phi}, \phi)^T = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} & \frac{i}{\sqrt{2}} \end{pmatrix}. \quad (6.27)$$

Then  $\langle \Psi, \Phi \rangle = \Psi \Phi = I$ .

### 6.3.4 Step 4: Get the Critical Eigenvalues

The eigenvalues of  $A$  are  $\pm i$ .

### 6.3.5 Step 5: Apply Orthogonal Decomposition

Let  $z(t, \mu)$  be a unique family of solutions of (6.21), where we will write  $z(t, \mu) = (z_1(t, \mu), z_2(t, \mu))^T$ . Now define

$$Y(t, \mu) = \langle \Psi, z(t, \mu) \rangle = \begin{pmatrix} \frac{z_1(t, \mu)}{\sqrt{2}} - \frac{iz_2(t, \mu)}{\sqrt{2}} \\ \frac{z_1(t, \mu)}{\sqrt{2}} + \frac{iz_2(t, \mu)}{\sqrt{2}} \end{pmatrix}. \quad (6.28)$$

If we let

$$B = \begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}, \quad B^* = \begin{pmatrix} -i & 0 \\ 0 & i \end{pmatrix}, \quad (6.29)$$

then

$$\frac{dY}{dt}(t, \mu) = BY(t, \mu) + \langle \Psi, f(z(t, \mu), \mu) \rangle, \quad (6.30)$$

or in an equivalent form

$$\begin{aligned} \frac{dy}{dt}(t, \mu) &= iy(t, \mu) + (\bar{\phi}, f(z(t, \mu), \mu)), \\ \frac{d\bar{y}}{dt}(t, \mu) &= -i\bar{y}(t, \mu) + (\phi, f(z(t, \mu), \mu)), \end{aligned} \quad (6.31)$$

where  $f(z(t, \mu), \mu) = (0, z_1(t, \mu)^2)^T$ .

We now develop the orthogonal function  $w(t, \mu)$  as

$$w(t, \mu) = z(t, \mu) - \Phi Y(t, \mu). \quad (6.32)$$

However, if we form  $\Phi Y(t, \mu)$  from (6.26) and (6.28) it is clear that  $\Phi Y(t, \mu) = (z_1(t, \mu), z_2(t, \mu))^T = z(t, \mu)$  and therefore from (6.32) that  $w(t, \mu) = 0$  as expected, since there are no other eigenvalues of  $A$  than  $i$  and  $-i$ . Thus (6.31) is the decomposed form of (6.21).

## 6.4 Delay Differential Equations as Operator Equations

As discussed earlier, the solutions to differential equations can be thought of in terms of trajectories or in terms of mappings of initial conditions. This same dichotomous point of view can be applied to DDEs. But, in the case of DDEs, the mapping or operator approach provides very fruitful qualitative results and, therefore, that approach will remain for this chapter.

### 6.4.1 Step 1: Form the Operator Equation

The principal difference between ordinary and delay differential equations is that, in ordinary differential equations, the initial condition space is finite dimensional and in DDEs it is infinite dimensional. The DDEs (6.1) and (6.2) can be thought of as maps of entire functions. In particular, we will start with the class of continuous functions defined on the interval  $[-\sigma, 0]$  with values in  $\mathbb{R}^n$  and refer to this as class  $C_0$ . The maps are constructed by defining a family of solution operators for the linear DDE (6.2) by

$$(T(t)\phi)(\theta) = (z_t(\phi))(\theta) = z(t + \theta) \quad (6.33)$$

for  $\phi \in C_0$ ,  $\theta \in [-\sigma, 0]$ ,  $s \geq 0$ . This is a mapping of a function in  $C_0$  to another function in  $C_0$ . Then (6.1) and (6.2) can be thought of as maps from  $C_0$  to  $C_0$ . The norm on the space is taken as

$$\|\phi\| = \max_{-\sigma \leq t \leq 0} |\phi(t)|, \quad (6.34)$$

where  $|\cdot|$  is the ordinary Euclidean 2-norm.

Figure 6.2 shows the mapping between an initial function in  $C_0$ ,  $\phi$ , to another function,  $z_t(\phi)$  in  $C_0$ .  $z_t(\phi)$  is the projection of the portion of the trajectory  $z(t)$  from  $t - \sigma$  to  $t$  back to  $C_0$ . This figure exhibits two approaches to looking at solutions of DDEs. One way is to consider a solution as a trajectory of  $z$  as a function of  $t$  with an initial condition function in  $C_0$ . The value of the solution at  $t$  on the trajectory graph depends on the values of the function on the trajectory from  $t - \sigma$  to  $t$ . Another way to look at the solutions of a DDE is to consider them as parameterized mappings  $T(t)\phi$  of functions  $\phi$  in  $C_0$ . From Fig. 6.2, this would be the portion of the trajectory from  $t - \sigma$  to  $t$  projected back to  $C_0$ . In the figure it is represented as  $z_t(\phi)$ , which is the function in  $C_0$  to which  $\phi$  is mapped under  $T(t)$ . The mapped function relates

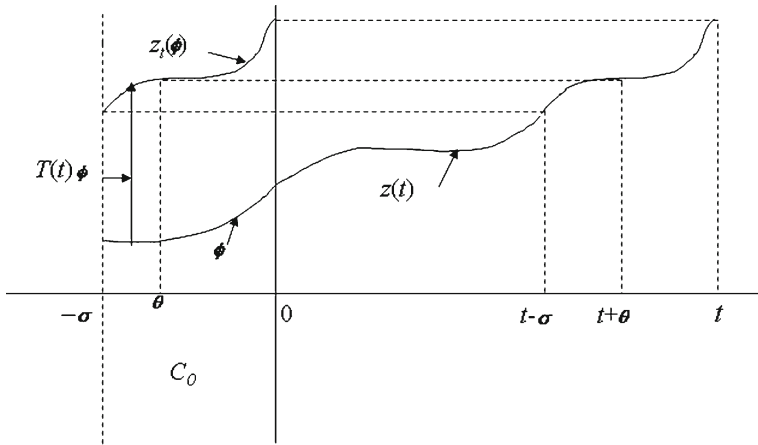


Fig. 6.2 The solution operator maps functions in  $C_0$  to functions in  $C_0$

to the trajectory as follows. The value of  $z_t(\phi)(\theta)$  for  $\theta \in [-\sigma, 0]$  is given as the trajectory value  $z(t + \theta)$ . This idea is not new, in that, in ordinary differential equations, solutions can be thought of in terms of either trajectories or maps of initial condition vectors in  $\mathbb{R}^n$ , say. Traditionally, we usually think of solving ordinary differential equations in terms of trajectories. For some qualitative analyses, however, the mapping approach is useful.

To determine what properties this operator must satisfy, we look at what basic properties that (6.5) satisfies. In particular, for each  $t$ ,  $T(t)$  is a bounded linear transformation for  $\phi \in \mathbb{R}^n$ . The boundedness comes from  $\|T(t)\phi\| \leq \|e^{At}\| \|\phi\|$ . For  $t > 0$ ,  $T(0)\phi = \phi$ , i.e.,  $T(0) = I$ . Finally,

$$\lim_{t \rightarrow t_0} \|T(t)\phi - T(t_0)\phi\| = 0, \quad (6.35)$$

since  $\|T(t)\phi - T(t_0)\phi\| \leq \|e^{A(t-t_0)}\| \|\phi\|$ .

Based on these properties, we formulate the following definition for a family of operators. A *Strongly Continuous Semigroup* satisfies

$$\begin{aligned} T(t) \text{ is bounded and linear for } t \geq 0, \\ T(0)\phi = \phi \text{ or } T(0) = I, \\ \lim_{t \rightarrow t_0} \|T(t)\phi - T(t_0)\phi\| = 0, \end{aligned} \quad (6.36)$$

where  $\|\cdot\|$  is an appropriate operator norm and  $\phi \in C_0$ . The family of operators,  $T(t)$ ,  $t \geq 0$ , is called a semigroup since the inverse property does not hold (see [18] and [35]).

If we take the derivative with respect to  $t$  in (6.5) we see that  $T(t)\phi$  satisfies (6.4). It is also easy to see that

$$A = \lim_{t \rightarrow 0} \frac{1}{t} (e^{At} - I). \quad (6.37)$$

We can call the matrix  $A$  the infinitesimal generator of the family  $T(t)$  in (6.5). The term infinitesimal generator can be thought of as arising from the formulation

$$dz = Az dt, \quad (6.38)$$

where for each infinitesimal increment,  $dt$ , in  $t$ ,  $A$  produces an infinitesimal increment,  $dz$ , in  $z$  at the point  $z$ .

In terms of operators, we define an operator called the infinitesimal generator. An *infinitesimal generator* of a semigroup  $T(t)$  is defined by

$$A\phi = \lim_{t \rightarrow 0^+} \frac{1}{t} [T(t)\phi - \phi], \quad (6.39)$$

for  $\phi \in C_0$ .

We will state the properties of the family of operators (6.33) without proofs, since the proofs require extensive knowledge of operator theory and they are not essential for the developments in this chapter. To begin with, the mapping (6.33) satisfies the semigroup properties on  $C_0$ . In the case of the linear system (6.2) the infinitesimal generator can be constructed as

$$(A(\mu)\phi) = \begin{cases} \frac{d\phi}{d\theta}(\theta) & -\sigma \leq \theta < 0, \\ U(\mu)\phi(0) + V(\mu)\phi(-\sigma) & \theta = 0, \end{cases} \quad (6.40)$$

where the parameter  $\mu$  is included in the definition of  $A$ . Then  $T(t)\phi$  satisfies

$$\frac{d}{dt} T(t)\phi = A(\mu)T(t)\phi, \quad (6.41)$$

where

$$\frac{d}{dt} T(t)\phi = \lim_{h \rightarrow 0} \frac{1}{h} (T(t+h) - T(t))\phi. \quad (6.42)$$

Finally, the operator form for the nonlinear DDE (6.1) can be written as

$$\frac{d}{dt} z_t(\phi) = A(\mu)z_t(\phi) + F(z_t(\phi), \mu), \quad (6.43)$$

where

$$(F(\phi, \mu))(\theta) = \begin{cases} 0 & -\sigma \leq \theta < 0, \\ f(\phi, \mu) & \theta = 0. \end{cases} \quad (6.44)$$

For  $\mu = 0$ , write  $f(\phi) = f(\phi, 0)$ ,  $F(\phi) = F(\phi, 0)$ . We note the analogy between the ordinary differential equation notation and the operator notation for the DDE.

### 6.4.2 Step 2: Define an Adjoint Operator

We can now construct a formal adjoint operator associated with (6.40). Let  $C_0^* = C([0, \sigma], \mathbb{R}^n)$  be the space of continuous functions from  $[0, \sigma]$  to  $\mathbb{R}^n$  with  $\|\psi\| = \max_{0 \leq \theta \leq \sigma} |\psi(\theta)|$  for  $\psi \in C_0^*$ . The formal adjoint equation associated with the linear DDE (6.2) is given by

$$\frac{du}{dt}(t, \mu) = -U(\mu)^T u(t, \mu) - V(\mu)^T u(t + \sigma, \mu). \quad (6.45)$$

If we define

$$(T^*(t)\psi)(\theta) = (u_t(\psi))(\theta) = u(t + \theta), \quad (6.46)$$

for  $\theta \in [0, \sigma]$ ,  $t \leq 0$ , and  $u_t \in C_0^*$ ,  $u_t(\psi)$  be the image of  $T^*(t)\psi$ , then (6.46) defines a strongly continuous semigroup with infinitesimal generator

$$(A^*(\mu)\psi) = \begin{cases} -\frac{d\psi}{d\theta}(\theta) & 0 < \theta \leq \sigma, \\ -\frac{d\psi}{d\theta}(0) = U(\mu)^T \psi(0) + V(\mu)^T \psi(\sigma) & \theta = 0. \end{cases} \quad (6.47)$$

Note that, although the formal infinitesimal generator for (6.46) is defined as

$$A_0^* \psi = \lim_{t \rightarrow 0^-} \frac{1}{t} [T^*(t)\psi - \psi], \quad (6.48)$$

Hale [12], for convenience, takes  $A^* = -A_0^*$  in (6.47) as the formal adjoint to (6.40). This family of operators (6.46) satisfies

$$\frac{d}{ds} T^*(t)\psi = -A^* T^*(t)\psi. \quad (6.49)$$

### 6.4.3 Step 3: Define a Natural Inner Product by Way of an Adjoint Operator

In contrast to  $R^n$ , the space  $C_0$  does not have a natural inner product associated with its norm. However, following Hale [12], one can introduce a substitute device that acts like an inner product in  $C_0$ . This is an approach that is often taken when a function space does not have a natural inner product associated with its norm. Spaces of functions that have natural inner products are called Hilbert spaces. Throughout, we will be assuming the complexification of the spaces so that we can work with complex eigenvalues and eigenvectors.

In analogy to (6.7) we start by constructing a Lagrange identity as follows. If

$$\begin{aligned} \Theta z(t) &= z'(t) - U(\mu)z(t) - V(\mu)z(t - \sigma), \\ \Omega u(t) &= u'(t) + U(\mu)^T u(t) + V(\mu)^T u(t + \sigma), \end{aligned} \quad (6.50)$$

then

$$\bar{u}^T(t)\Theta z(t) + \overline{\Omega u}^T(t)z(t) = \frac{d}{dt} \langle u, z \rangle(t), \quad (6.51)$$

where

$$\langle u, z \rangle(t) = \bar{u}^T(t)z(t) + \int_{t-\sigma}^t \bar{u}^T(s + \sigma)V(\mu)z(s)ds. \quad (6.52)$$

Deriving the natural inner product for  $R^n$  from the Lagrange identity (6.7) motivates the derivation of (6.52). Again, if  $z$  and  $u$  satisfy  $\Theta z(t) = 0$  and  $\Omega u(t) = 0$  then, from (6.52),  $(d/dt)\langle u, z \rangle(t) = 0$ , which implies  $\langle u, z \rangle(t)$  is constant and one can set  $t = 0$  in (6.52) and define the form

$$\langle u, z \rangle = \bar{u}^T(0)z(0) + \int_{-\sigma}^0 \bar{u}^T(s + \sigma)V(\mu)z(s)ds. \quad (6.53)$$

One can now state some properties of (6.40), (6.47), and (6.53) that are analogs of the properties given in Sect. 6.2 for ordinary differential equations.

1. For  $\phi \in C_0$ ,  $\psi \in C_0^*$ ,

$$\langle \psi, A(\mu)\phi \rangle = \langle A^*(\mu)\psi, \phi \rangle. \quad (6.54)$$

2.  $\lambda$  is an eigenvalue of  $A(\mu)$  if and only if  $\bar{\lambda}$  is an eigenvalue of  $A^*(\mu)$ .
3. The dimensions of the eigenspaces of  $A(\mu)$  and  $A^*(\mu)$  are finite and equal.
4. If  $\psi_1, \dots, \psi_d$  is a basis for the right eigenspace of  $A^*(\mu)$  and the associated  $\phi_1, \dots, \phi_d$  is a basis for the right eigenspace of  $A(\mu)$ , construct the matrices  $\Psi = (\psi_1, \dots, \psi_d)$  and  $\Phi = (\phi_1, \dots, \phi_d)$ . Define the bilinear form between  $\Psi$  and  $\Phi$  by

$$\langle \Psi, \Phi \rangle = \begin{pmatrix} \langle \psi_1, \phi_1 \rangle & \dots & \langle \psi_1, \phi_d \rangle \\ \vdots & \ddots & \vdots \\ \langle \psi_d, \phi_1 \rangle & \dots & \langle \psi_d, \phi_d \rangle \end{pmatrix}. \quad (6.55)$$

This matrix is nonsingular and can be chosen so that  $\langle \Psi, \Phi \rangle = I$ . Note that if (6.55) is not the identity then a change of coordinates can be performed by setting  $K = \langle \Psi, \Phi \rangle^{-1}$  and  $\Phi' = \Phi K$ . Then  $\langle \Psi, \Phi' \rangle = \langle \Psi, \Phi K \rangle = \langle \Psi, \Phi \rangle K = I$ . Equation (6.55) also satisfies the inner product properties (6.9).

#### 6.4.4 Step 4: Get the Critical Eigenvalues

The eigenvalues for (6.40) are given by the  $\lambda$  solutions of the transcendental equation

$$\det(\lambda I - U(\mu) - e^{-\lambda\sigma}V(\mu)) = 0. \quad (6.56)$$

This form of characteristic equation, sometimes called an *exponential polynomial*, has been studied in Avellar and Hale [4], Bellman and Cooke [7], Hale and Lunel [15], Kuang [23], and Pinney [27]. The solutions are called the *eigenvalues* of (6.2) and, in general, there are an infinite number of them. For a discussion of the general expansion of solutions of (6.2) in terms of the eigenvalues see Bellman and Cooke [7] or Pinney [27]. The actual computation of these eigenvalues can become very involved as the reader will see in the example that will be considered later in this chapter. Here, though, we will only be concerned with conditions for the existence of eigenvalues of the form  $i\omega$  and  $-i\omega$  and we further limit ourselves to the case in which there are only two eigenvalues  $i\omega$  and  $-i\omega$  and all other eigenvalues have negative real parts. The significance of this is that we will be looking for conditions

for which the family of eigenvalues, as a function of the parameter  $\mu$ , passes across the imaginary axis. These conditions will be the Hopf conditions referred to earlier in this chapter. The value of  $\omega$  is related to the natural frequency of oscillation of the linear part of the DDE system.

### 6.4.5 Step 5: Apply Orthogonal Decomposition

For the sake of notation, let  $A = A(\mu)$ ,  $A^* = A^*(\mu)$ ,  $U = U(\mu)$ ,  $V = V(\mu)$ ,  $\omega = \omega(\mu)$ . The basis eigenvectors for  $A$  and  $A^*$  associated with the eigenvalues  $\lambda = i\omega$ ,  $\bar{\lambda} = -i\omega$  will be denoted as  $\phi_C$ ,  $\bar{\phi}_C$  and  $\phi_D$ ,  $\bar{\phi}_D$ , respectively, where the subscripts  $C$  and  $D$  refer to parameters defining the basis vectors and depend on  $U$  and  $V$ .

We define the matrix

$$\Phi = (\phi_C, \bar{\phi}_C). \quad (6.57)$$

The two eigenvectors for  $A$ , associated with the eigenvalues  $\lambda = i\omega$ ,  $\bar{\lambda} = -i\omega$ , are given by

$$\begin{aligned} \phi_C(\theta) &= e^{i\omega\theta} C, \\ \bar{\phi}_C(\theta) &= e^{-i\omega\theta} \bar{C}, \end{aligned} \quad (6.58)$$

where  $C$  is a  $2 \times 1$  vector. With these functions defined, it is clear that  $\Phi$  is a function of  $\theta$  and should formally be written as  $\Phi(\theta)$ . Note that  $\Phi(0) = (C, \bar{C})$ . However, in order to simplify notation we will write  $\Phi = \Phi(\theta)$  but we will sometimes refer to  $\Phi(0)$ . These functions follow from (6.40). If  $-\sigma \leq \theta < 0$  then  $d\phi/d\theta = i\omega\phi$  implies  $\phi(\theta) = \exp(i\omega\theta)C$  where  $C = (c_1, c_2)^T$ . For  $\theta = 0$ , (6.40) implies  $(U + V \exp(-i\omega\sigma))C = i\omega C$  or  $(U - i\omega I + V \exp(-i\omega\sigma))C = 0$ . Since  $i\omega$  is an eigenvalue, (6.56) implies that there is a nonzero solution  $C$ .

Similarly, the eigenvectors for  $A^*$  associated with the eigenvalues  $-i\omega, i\omega$  are also given by

$$\begin{aligned} \phi_D(\theta) &= e^{i\omega\theta} D, \\ \bar{\phi}_D(\theta) &= e^{-i\omega\theta} \bar{D}, \end{aligned} \quad (6.59)$$

where  $D = (d_1, d_2)^T$ . Again, this follows from (6.47) since, from  $0 < \theta \leq \sigma$ ,

$$-\frac{d\phi}{d\theta} = -i\omega\phi, \quad (6.60)$$

we can compute the solutions given in (6.59). Define the matrix

$$\Psi = (\phi_D, \bar{\phi}_D) \quad (6.61)$$

where  $D$  is computed as follows. At  $\theta = 0$  we have from (6.47) that

$$\left( U^T + V^T e^{i\omega\sigma} + i\omega I \right) D = 0. \quad (6.62)$$

The determinant of the matrix on the left is the characteristic equation so that there is a nonzero  $D$ .

From (6.57), (6.58), (6.59), and (6.61) one seeks to solve for  $D$  so that

$$\langle \Psi, \Phi \rangle = \begin{pmatrix} \langle \phi_D, \phi_C \rangle & \langle \phi_D, \bar{\phi}_C \rangle \\ \langle \bar{\phi}_D, \phi_C \rangle & \langle \bar{\phi}_D, \bar{\phi}_C \rangle \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \quad (6.63)$$

Due to symmetry we only need to satisfy  $\langle \phi_D, \phi_D \rangle = 1$  and  $\langle \phi_D, \bar{\phi}_D \rangle = 0$ .

On eigenspaces, the infinitesimal generators can be represented by matrices. In fact  $A$  and  $A^*$  satisfy  $AQ = QB$ ,  $A^*\Psi = \Psi B^*$  where  $\Phi$ ,  $\Psi$  are given by (6.57) and (6.61) and the matrices  $B, B^*$  are also given by (6.12).

Now that we have constructed the adjoint and given some of its properties we can decompose the nonlinear operator equation (6.43) into a two-dimensional system with eigenvalues  $i\omega$  and  $-i\omega$  and another operator equation with eigenvalues having negative real parts. The procedure is based on Hale [12] and is similar to Step 5 of Section 6.2.

We will decompose the nonlinear system (6.43) for the case  $\mu = 0$ , since we will not need to develop approximations for  $\mu \neq 0$  in this chapter. High order approximations have been developed in Hassard and Wan [16], but these will not be needed in order to develop the approximate bifurcating periodic solution studied here.

Based on standard existence and uniqueness theorems for DDEs, let  $z_t \in C_0$  be the unique family of solutions of (6.43), where the  $\mu$  notation has been dropped since we are only working with  $\mu = 0$ . Define

$$Y(t) = \langle \Psi, z_t \rangle = \begin{pmatrix} \langle \phi_D, z_t \rangle \\ \langle \bar{\phi}_D, z_t \rangle \end{pmatrix}, \quad (6.64)$$

where  $Y(t) \in C^2$  for  $t \geq 0$ , and set  $Y(t) = (y(t), \bar{y}(t))^T$  where  $y(t) = \langle \phi_D, z_t \rangle$  and  $\bar{y}(t) = \langle \bar{\phi}_D, z_t \rangle$ . The reader should note the similarity to (6.11).

By differentiating (6.53), and using  $z_t(0) = z(t)$ ,  $z_t(\theta) = z(t + \theta)$ , we have

$$\frac{d}{dt} \langle \Psi, z_t \rangle = \langle \Psi, \frac{dz_t}{dt} \rangle. \quad (6.65)$$

Use (6.9), (6.43), (6.54), (6.65), and  $A^*\Psi = \Psi B^*$  to write

$$\frac{d}{dt} Y(t) = BY(t) + \langle \Psi, F(z_t) \rangle. \quad (6.66)$$

Using (6.44) and (6.53), compute  $\langle \phi_D, F(z_t) \rangle = \bar{\phi}_D^T(0)(F(z_t))(0) = \bar{D}^T f(z_t)$ . Similarly  $\langle \bar{\phi}_D, F(z_t) \rangle = D^T f(z_t)$ . Then

$$\langle \Psi, F(z_t) \rangle = \begin{pmatrix} \langle \phi_D, F(z_t) \rangle \\ \langle \bar{\phi}_D, F(z_t) \rangle \end{pmatrix} = \begin{pmatrix} \bar{D}^T f(z_t) \\ D^T f(z_t) \end{pmatrix}, \quad (6.67)$$

which yields the first two equations in (6.83). They can be written as

$$\begin{aligned}\frac{d}{dt}y(t) &= i\omega y(t) + \bar{D}^T f(z_t), \\ \frac{d}{dt}\bar{y}(t) &= -i\omega \bar{y}(t) + D^T f(z_t).\end{aligned}\quad (6.68)$$

If one defines the orthogonal family of functions

$$w_t = z_t - \Phi Y(t), \quad (6.69)$$

where  $\Phi$  is given by (6.57) and  $Y(t)$  is given by (6.64), then  $\langle \Psi, w_t \rangle = 0$ , where  $\Psi$  is given by (6.61). Now apply the infinitesimal generator (6.40) to

$$z_t = w_t + \Phi Y(t), \quad (6.70)$$

to get

$$(Az_t)(\theta) = (Aw_t)(\theta) + (A\Phi)(\theta)Y(t) = (Aw_t)(\theta) + \Phi BY(t). \quad (6.71)$$

We will need this relation below.

One can now construct the third equation. There are two cases:  $\theta = 0$  and  $\theta \in [-\sigma, 0)$ . From (6.69), for the case with  $\theta = 0$ ,

$$w(t) = w_t(0) = z_t(0) - \Phi(0)Y(t) = z(t) - \Phi(0)Y(t). \quad (6.72)$$

The reader is reminded here of the notation  $w_t(\theta) = w(t + \theta)$ . It is easy to show that  $W(t) \in R^2$ , since  $x(t) \in R^2$  and  $\Phi(0)Y(t) = \langle \phi_D, z_t \rangle C + \langle \bar{\phi}_D, z_t \rangle \bar{C} = 2\text{Re}\{\langle \phi_D, z_t \rangle C\} \in R^2$ .

From (6.40) and (6.43)

$$\frac{d}{dt}z(t) = \frac{dz_t}{dt}(0) = (Az_t)(0) + (F(z_t))(0) = Uz(t) + Vz(t - \sigma) + f(z_t). \quad (6.73)$$

Differentiate (6.72) and combine it with (6.66) and (6.73) to give

$$\frac{d}{dt}w(t) = \{Uz(t) + Vz(t - \sigma) + f(z_t)\} - \Phi(0)\{BY(t) - \langle \Psi, F(z_t) \rangle\}. \quad (6.74)$$

If  $\theta = 0$  in (6.71) then, from (6.40),

$$Uz(t) + Vz(t - \sigma) = Uw(t) + Vw(t - \sigma) + \Phi(0)BY(t). \quad (6.75)$$

Now substitute (6.75) into (6.74) to get

$$\begin{aligned}\frac{d}{dt}w(t) &= Uw(t) + Vw(t - \sigma) + f(z_t) - \Phi(0)\langle \Psi, F(z_t) \rangle, \\ &= Uw(t) + Vw(t - \sigma) + f(z_t) - 2\text{Re}\{\langle \phi_D, z_t \rangle C\}.\end{aligned}\quad (6.76)$$

For the case with  $\theta \neq 0$  we can apply a similar argument to that used to create (6.76). We start by differentiating (6.69) to get

$$\begin{aligned}\frac{dw_t}{dt} &= \frac{dz_t}{dt} - \Phi Y'(t), \\ &= \frac{dz_t}{dt} - \Phi \{BY(t) + \langle \Psi, F(z_t) \rangle\}, \\ &= \frac{dz_t}{dt} - \Phi BY(t) - \Phi \langle \Psi, F(z_t) \rangle.\end{aligned}\tag{6.77}$$

For  $\theta \neq 0$  in (6.43) and (6.44)

$$\frac{dz_t}{dt} = Az_t.\tag{6.78}$$

Then, using (6.71) and (6.78), we have

$$\begin{aligned}\frac{dw_t}{dt} &= Az_t - \Phi BY(t) - \Phi \langle \Psi, F(z_t) \rangle, \\ &= Aw_t + \Phi BY(t) - \Phi BY(t) - \Phi \langle \Psi, F(z_t) \rangle,\end{aligned}\tag{6.79}$$

$$= Aw_t - \Phi \langle \Psi, F(z_t) \rangle.\tag{6.80}$$

This can then be written as

$$\frac{dw_t}{ds} = Aw_t - 2\text{Re}\{\langle \phi_D, z_t \rangle \phi_C\}.\tag{6.81}$$

Use (6.44) to finally write the equation

$$\frac{dw_t}{dt} = Aw_t - 2\text{Re}\{\langle \phi_D, z_t \rangle \phi_C\} + F(z_t).\tag{6.82}$$

Equation (6.43) has now been decomposed as

$$\begin{aligned}\frac{d}{dt}y(t) &= i\omega y(t) + \overline{D}^T f(z_t), \\ \frac{d}{dt}\bar{y}(t) &= -i\omega \bar{y}(t) + D^T f(z_t), \\ \frac{d}{dt}w_t(\theta) &= \begin{cases} (Aw_t)(\theta) - 2\text{Re}\{\langle \phi_D, z_t \rangle \phi_C(\theta) \} & -\sigma \leq \theta < 0, \\ (Aw_t)(0) - 2\text{Re}\{\langle \phi_D, z_t \rangle \phi_C(0) \} + f(z_t) & \theta = 0. \end{cases}\end{aligned}\tag{6.83}$$

In order to simplify the notation write (6.83) in the form

$$\begin{aligned}\frac{dy}{dt}(t) &= i\omega y(t) + F_1(Y, w_t), \\ \frac{d\bar{y}}{dt}(t) &= -i\omega \bar{y}(t) + \overline{F}_1(Y, w_t), \\ \frac{dw_t}{dt} &= Aw_t + F_2(Y, w_t),\end{aligned}\tag{6.84}$$

where

$$\begin{aligned} F_1(Y, w_t) &= \bar{D}^T f(z_t), \\ F_2(Y, w_t) &= \begin{cases} -2\text{Re}\{\langle \phi_D, z_t \rangle \phi_C(\theta)\} & -\sigma \leq \theta < 0, \\ -2\text{Re}\{\langle \phi_D, z_t \rangle \phi_C(0)\} + f(z_t) & \theta = 0. \end{cases} \end{aligned} \quad (6.85)$$

Note that (6.84) is a coupled system. The center manifold and normal forms will be used as a tool to partially decouple this system.

## 6.5 A Machine Tool DDE Example: Part 1

This example will be discussed in multiple parts. In the first part, we will formulate the operator form for the example of DDE, describe a process of determining the critical eigenvalues for the problem, and formulate the adjoint operator equation.

### 6.5.1 Step 1: Form the Operator Equation

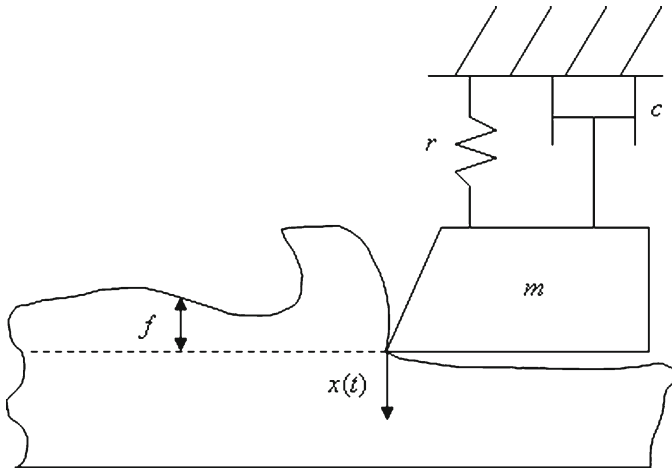
The example we will consider involves a turning center and workpiece combination. For readers unfamiliar with turning centers they can be thought of as numerically controlled lathes. The machine tool model, used only for illustration in this chapter, is taken from Kalmár-Nagy et al. [20] and can be written as

$$\ddot{x} + 2\xi\omega_n\dot{x} + \omega_n^2x = \frac{kf_0}{m\alpha} \left( 1 - \left( \frac{f}{f_0} \right)^\alpha \right), \quad (6.86)$$

where  $\omega_n = \sqrt{r/m}$  is the natural frequency of the undamped free oscillating system and  $\xi = c/2m\omega_n$  is the relative damping factor and  $k$  is the cutting force coefficient that is related to the slope of the power-law curve used to define the right-hand side of (6.86). The parameters  $m$ ,  $r$ ,  $c$ , and  $\alpha$  are taken as  $m = 10\text{kg}$ ,  $r = 3.35\text{MNm}^{-1}$ ,  $c = 156\text{kg s}^{-1}$ , and  $\alpha = 0.41$  and were obtained from measurements of the machine–tool response function (see Kalmár-Nagy et al. [20]). The parameter  $\alpha$  was obtained from a cutting force model in Taylor [31]. These then imply that  $\omega_n = 578.791/\text{s}$ ,  $\xi = 0.0135$ . The nominal chip width is taken as  $f_0$  and the time varying chip width is

$$f = f_0 + x(t) - x(t - \tau), \quad (6.87)$$

where the delay  $\tau = 2\pi/\Omega_\tau$  is the time for one revolution of the turning center spindle. The cutting force parameter  $k$  will be taken as the bifurcation parameter since we will be interested in the qualitative change in the displacement,  $x(t)$ , as the cutting force changes. The parameters  $m$ ,  $r$ ,  $c$ , and  $f$  are shown in Fig. 6.3. The displacement  $x(t)$  is directed positively into the workpiece and the tool is assumed not to leave the workpiece.



**Fig. 6.3** One degree-of-freedom-model for single-point turning

The model is simplified by introducing a nondimensional time  $s$  and displacement  $z$  by

$$\begin{aligned} s &= \omega_n t, \\ z &= \frac{x}{A}, \end{aligned} \quad (6.88)$$

where the lengthscale is computed as

$$A = \frac{3f_0}{2 - \alpha}, \quad (6.89)$$

a new bifurcation parameter  $p$  is set to

$$p = \frac{k}{m\omega_n^2} \quad (6.90)$$

and the delay parameter becomes

$$\sigma = \omega_n \tau. \quad (6.91)$$

The dimensionless model then becomes, after expanding the right-hand side of (6.86) to the third order,

$$\frac{d^2x}{ds^2} + 2\xi \frac{dx}{ds} + x = p (\Delta x + E(\Delta x^2 + \Delta x^3)), \quad (6.92)$$

where

$$\begin{aligned} \Delta x &= x(s - \sigma) - x(s), \\ E &= \frac{3(1 - \alpha)}{2(2 - \alpha)}. \end{aligned} \quad (6.93)$$

We will now consider  $x$  as a function of the dimensionless  $s$  instead of  $t$ . The linear part of the model is given by

$$\frac{d^2x}{ds^2} + 2\xi \frac{dx}{ds} + x = p\Delta x, \quad (6.94)$$

Since the Hopf bifurcation studied in this chapter is local, the bifurcation parameter will be written as

$$p = \mu + p_c, \quad (6.95)$$

where  $p_c$  is a critical value at which bifurcation occurs. Then (6.92) can be put into vector form (6.1) by letting  $z_1(s) = x(s)$ ,  $z_2(s) = x'(s)$ . Then

$$\frac{dz}{ds}(s) = U(\mu)z(s) + V(\mu)z(s - \sigma) + f(z(s), z(s - \sigma), \mu), \quad (6.96)$$

where

$$\begin{aligned} z(s) &= \begin{pmatrix} z_1(s) \\ z_2(s) \end{pmatrix}, \\ U(\mu) &= \begin{pmatrix} 0 & 1 \\ -1 - (\mu + p_c) & -2\xi \end{pmatrix}, \\ V(\mu) &= \begin{pmatrix} 0 & 0 \\ \mu + p_c & 0 \end{pmatrix}, \end{aligned} \quad (6.97)$$

and

$$\begin{aligned} f(z(s), z(s - \sigma), \mu) & \quad (6.98) \\ &= \begin{pmatrix} 0 \\ (\mu + p_c)E(z_1(s - \sigma) - z_1(s))^2 + (\mu + p_c)E(z_1(s - \sigma) - z_1(s))^3 \end{pmatrix}. \end{aligned}$$

The linear portion of this equation is given by

$$\frac{dz}{ds}(s) = U(\mu)z(s) + V(\mu)z(s - \sigma) \quad (6.99)$$

and the infinitesimal generator is given by

$$(A(\mu)\phi) = \begin{cases} \frac{d\phi}{d\theta}(\theta) & -\sigma \leq \theta < 0 \\ U(\mu)\phi(0) + V(\mu)\phi(-\sigma) & \theta = 0 \end{cases}. \quad (6.100)$$

Then (6.96) can easily be put into the operator form (6.43).

### 6.5.2 Step 2: Define the Adjoint Operator

Here we can follow the lead of Step 2 of Sect. 6.4 and define the formal adjoint as

$$\frac{dz}{ds}(s, \mu) = -U(\mu)^T z(s, \mu) - V(\mu)^T z(s + \sigma, \mu). \quad (6.101)$$

As in Sect. 6.4, if we define

$$(T^*(s)\psi)(\theta) = (z_s(\psi))(\theta) = z(s + \theta) \quad (6.102)$$

for  $\theta \in [0, \sigma]$ ,  $s \leq 0$ ,  $u_s \in C_0^*$ , and  $z_s(\psi)$  as the image of  $T^*(s)\psi$ , then (6.102) defines a strongly continuous semigroup with infinitesimal generator

$$(A^*(\mu)\psi) = \begin{cases} -\frac{d\psi}{d\theta}(\theta) & 0 < \theta \leq \sigma, \\ -\frac{d\psi}{d\theta}(0) = U(\mu)^T\psi(0) + V(\mu)^T\psi(\sigma) & \theta = 0. \end{cases} \quad (6.103)$$

Note that, although, as before, the formal infinitesimal generator for (6.102) is defined as

$$A_0^*\psi = \lim_{s \rightarrow 0^-} \frac{1}{s} [T^*(s)\psi - \psi]. \quad (6.104)$$

Hale [12], for convenience, takes  $A^* = -A_0^*$  in (6.103) as the formal adjoint to (6.100).

### 6.5.3 Step 3: Define a Natural Inner Product by Way of an Adjoint

This step follows simply by defining the inner product in the same manner as in (6.53).

### 6.5.4 Step 4: Get the Critical Eigenvalues

In this step, the reader will begin to see some of the complexity of dealing with the transcendental characteristic equation. The eigenvalues will depend on the parameters in (6.94) and only certain parameter combinations will lead to eigenvalues of the form  $i\omega$  and  $-i\omega$ . We will also establish the connection of the critical eigenvalues with the Hopf bifurcation conditions.

Following Hale [14], introduce the trial solution

$$z(s) = ce^{\lambda s}, \quad (6.105)$$

where  $c \in C^2$ , and  $U(\mu)$  and  $V(\mu)$  are given by (6.97), into the linear system (6.94) and set the determinant of the resulting system to zero. This yields the transcendental characteristic equation

$$\chi(\lambda) = \lambda^2 + 2\xi\lambda + (1 + p) - pe^{-\lambda\sigma} = 0. \quad (6.106)$$

Before developing the families of conjugate eigenvalues, we wish to characterize certain critical eigenvalues of (6.106) of the form  $\lambda = i\omega$ . However, the eigenvalues for (6.106) of the form  $\lambda = i\omega$  exist only for special combinations of  $p$  and  $\sigma$ . We will say that a triple  $(\omega, \sigma, p)$ , where  $\omega, \sigma, p$  are real, will be called a *critical eigen*

*triple* of (6.106) if  $\lambda = i\omega$ ,  $\sigma$ ,  $p$  simultaneously satisfy (6.106). The discussion below points out the significant computational difficulties involved with estimating the eigenvalues for a characteristic equation or exponential polynomial related to a linear DDE.

The following properties characterize the critical eigen triples for linear delay equations of the form (6.2) with coefficients from (6.97):

1.  $(\omega, \sigma, p)$  is a critical eigen triple of (6.106) if and only if  $(-\omega, \sigma, p)$  also is a critical eigen triple of (6.106).
2. For  $\omega > 1$  there is a uniquely defined sequence  $\sigma_r = \sigma_r(\omega)$ ,  $r = 0, 1, 2, \dots$ , and a uniquely defined  $p = p(\omega)$  such that  $(\omega, \sigma_r, p)$ ,  $r = 0, 1, 2, \dots$ , are critical eigen triples.
3. If  $(\omega, \sigma, p)$  is a critical eigen triple, with  $\omega > 1$ , then  $p \geq 2\xi(1 + \xi)$ . That is, no critical eigen triple for (6.106) exists for  $p < 2\xi(1 + \xi)$ .
4. For

$$p_m = 2\xi(1 + \xi), \quad (6.107)$$

the minimum  $p$  value, there is a unique  $\omega > 1$  and a unique sequence  $\sigma_r$ ,  $r = 0, 1, 2, \dots$ , such that  $(\omega_m, \sigma_r, p_m)$  is a critical eigen triple for (6.106) for  $r = 0, 1, 2, \dots$ . The frequency at the minimum is

$$\omega_m = \sqrt{1 + 2\xi}. \quad (6.108)$$

5. For  $p > 2\xi(1 + \xi)$  there exist two  $\omega$ s,  $\omega > 1$ , designated  $\omega_+$ ,  $\omega_-$  and uniquely associated sequences  $\sigma_r^+ = \sigma_r(\omega_+)$ ,  $\sigma_r^- = \sigma_r(\omega_-)$ ,  $r = 0, 1, 2, \dots$  such that  $(\omega_+, \sigma_r^+, p)$ ,  $(\omega_-, \sigma_r^-, p)$  are critical eigen triples for (6.106) for  $r = 0, 1, 2, \dots$ .  $\omega_+$ ,  $\omega_-$  are given by

$$\omega_+^2 = (1 + p - 2\xi^2) + \sqrt{p^2 - 4\xi^2 p + (4\xi^4 - 4\xi^2)}, \quad (6.109)$$

$$\omega_-^2 = (1 + p - 2\xi^2) - \sqrt{p^2 - 4\xi^2 p + (4\xi^4 - 4\xi^2)}. \quad (6.110)$$

$\sigma_r^+$ ,  $\sigma_r^-$  are given by

$$\sigma_r^+ = \frac{2(\psi_+ + r\pi) + 3\pi}{\omega_+}, \quad (6.111)$$

$$\sigma_r^- = \frac{2(\psi_- + r\pi) + 3\pi}{\omega_-}, \quad (6.112)$$

where

$$\psi_+ = -\pi + \tan^{-1} \left( \frac{2\xi\omega_+}{\omega_+^2 - 1} \right), \quad (6.113)$$

$$\psi_- = -\pi + \tan^{-1} \left( \frac{2\xi\omega_-}{\omega_-^2 - 1} \right), \quad (6.114)$$

6. There do not exist critical eigen triples for  $0 \leq \omega \leq 1$ .

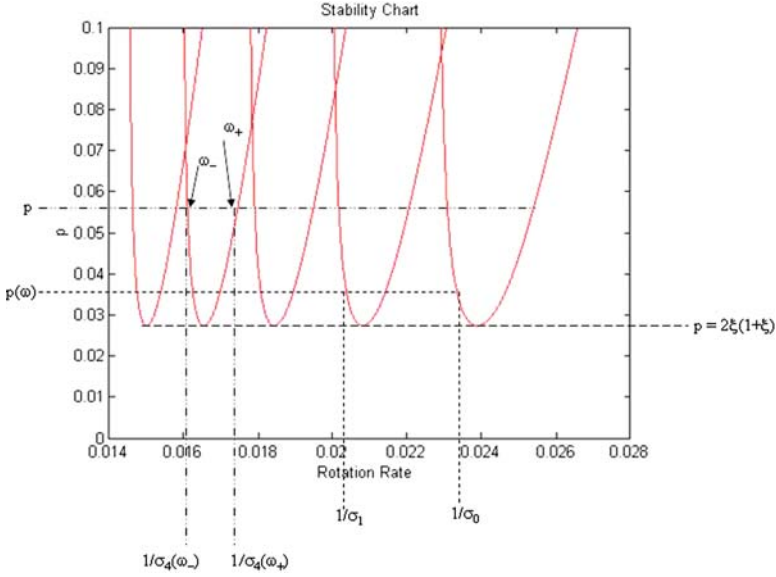


Fig. 6.4 Stability chart with sample critical eigen triples identified

We will not prove these results (for proofs see Gilsinn [11]) but we briefly discuss their significance graphically by examining Fig. 6.4 where the plots are based on the value of  $\xi = 0.0135$ . The entire development of the periodic solutions on the center manifold depends on knowing the critical bifurcation parameter  $p$  in (6.92). This parameter is linked to the rotation rate,  $\Omega_r$ , of the turning center spindle. One can plot  $p$  against  $\Omega_r = 1/\sigma_r$ , where  $\Omega_r$  is the rotation rate of the turning center spindle, for  $r = 0, 1, 2, \dots$  where each  $r$  indexes a lobe in Fig. 6.4 moving from right to left in the figure. Call the right most lobe, lobe 0, the next on the left lobe 1, etc. For each  $r$  the pairs  $(\Omega_r, p)$  are computed for a vector of  $\omega$  values. When these families of pairs are plotted they form a family of  $N$  lobes. Each lobe is parameterized by the same vector of  $\omega$ s so that each point on a lobe boundary represents an eigenvalue of (6.106) for a given  $p$  and  $\sigma_r = 1/\Omega_r$ . The minimum of each lobe is asymptotic to a line often called the *stability limit*. The second property above states that for a given  $\omega$  there is associated a unique value on the vertical axis, called  $p(\omega)$ , but an infinite number of  $\sigma_r(\omega)$ s, one for each lobe, depicted graphically on the horizontal axis as  $1/\omega_r$ , for rotation rate. The minimum value on each lobe occurs at  $p = 2\xi(1 + \xi)$  with an associated unique  $\omega = \sqrt{1 + 2\xi}$ . Finally, for each lobe there are two  $\omega$ s associated with each  $p$ , denoted by  $\omega_-$  and  $\omega_+$ , where  $\omega_-$  is the parameter associated with the left side of the lobe and  $\omega_+$  with the right side of the lobe. At the minimum  $\omega_- = \omega_+$ .

The significance of the stability chart is that the lobe boundaries divide the plane into regions of stable and unstable machining. In particular, the regions below

the lobes are considered stable and those above are considered unstable in the manufacturing sense. This will be a result of the Hopf bifurcation at the lobe boundaries. Since the parameter  $p$  is proportional to material removal, the regions between lobes represent areas that can be exploited for material removal above the stability limit line. This figure, called a *stability chart*, graphically shows the meanings of properties (2)–(5) above and was introduced by Tobias and Fishwick [33]. The structure of stability charts for DDE can be very complex. The current machine tool example exhibits one of the simpler ones. To see some examples of different stability charts the reader is referred to the book by Stépán [28].

We will use an argument modeled after Altintas and Budak [1] to develop necessary conditions for  $\sigma_r$  and  $p$  and show how they relate to  $\omega$ . These conditions are in fact used to graphically display the stability lobes. Set

$$\Phi(\lambda) = \frac{1}{\lambda^2 + 2\xi\lambda + 1}. \quad (6.115)$$

Then (6.106) becomes

$$1 + p(1 - e^{-\lambda\sigma})\Phi(\lambda) = 0. \quad (6.116)$$

Set  $\lambda = i\omega$  and write

$$\Phi(i\omega) = G(\omega) + iH(\omega), \quad (6.117)$$

where

$$G(\omega) = \frac{1 - \omega^2}{(1 - \omega^2)^2 + (2\xi\omega)^2}, \quad (6.118)$$

$$H(\omega) = \frac{-2\xi\omega}{(1 - \omega^2)^2 + (2\xi\omega)^2}. \quad (6.119)$$

Substitute (6.117) into (6.116) and separate real and imaginary parts to get

$$1 + p[(1 - \cos \omega\sigma)G(\omega) - (\sin \omega\sigma)H(\omega)] = 0, \quad (6.120)$$

$$p[G(\omega) \sin \omega\sigma + H(\omega)(1 - \cos \omega\sigma)] = 0. \quad (6.121)$$

From (6.118) and (6.119)

$$\frac{H(\omega)}{G(\omega)} = -\frac{\sin \omega\sigma}{1 - \cos \omega\sigma}. \quad (6.122)$$

From the definition of  $G$ ,  $H$ , and the fact that  $\omega > 1$ , (6.122) falls in the third quadrant so that one can introduce the phase angle for (6.117), using (6.118) and (6.119), as

$$\psi = \tan^{-1} \left( \frac{H(\omega)}{G(\omega)} \right) = -\pi + \tan^{-1} \left( \frac{2\xi\omega}{\omega^2 - 1} \right). \quad (6.123)$$

Clearly,  $-\pi \leq \psi \leq \pi$ . Using half-angle formulas,

$$\begin{aligned}
 \tan \psi &= -\frac{\sin \omega \sigma}{1 - \cos \omega \sigma}, \\
 &= -\frac{\cos\left(\frac{\omega \sigma}{2}\right)}{\sin\left(\frac{\omega \sigma}{2}\right)}, \\
 &= -\cot\left(\frac{\omega \sigma}{2}\right), \\
 &= \tan\left(\frac{\pi}{2} + \frac{\omega \sigma}{2} \pm n\pi\right),
 \end{aligned} \tag{6.124}$$

for  $n = 0, 1, 2, \dots$ . Therefore

$$\psi = \frac{\pi}{2} + \frac{\omega \sigma}{2} \pm n\pi, \tag{6.125}$$

where  $\omega \sigma > 0$  must be satisfied for all  $n$ . In order to satisfy this and the condition that  $-\pi \leq \psi \leq \pi$ , select the negative sign and

$$n = 2 + r, \tag{6.126}$$

for  $r = 0, 1, 2, \dots$ . Therefore, from (6.125), the necessary sequence,  $\sigma_r$ , is given by

$$\sigma_r = \frac{\omega}{2(\psi + r\pi) + 3\pi}, \tag{6.127}$$

where  $\psi$  is given by (6.123). Finally, substituting (6.122) into (6.120), one has the necessary condition for  $p$  as

$$p = -\frac{1}{2G(\omega)}, \tag{6.128}$$

where,  $p > 0$  since  $\omega > 1$ . Therefore (6.127) and (6.128) are the necessary conditions for  $(\omega, \sigma_r, p)$ ,  $r = 0, 1, 2, \dots$ , to be critical eigen triples for (6.106). Note that this also implies uniqueness. Equations (6.127) and (6.128) show how  $p = p(\omega)$  and  $1/\sigma_r$  uniquely relate in Fig. 6.4.

The lobes in Fig. 6.4 are plotted by the following algorithm. Since  $p$  must be positive, select any set of values  $\omega > 1$  such that  $G(\omega) < 0$ . Given a set of  $\omega > 1$  values, pick  $r = 0, 1, 2, \dots$  for as many lobes as desired. Compute  $1/\sigma_r$  from (6.127) and  $p$  from (6.128), then plot the pairs  $(1/\sigma_r, p)$ .

The following is a consequence of properties (1)–(6) for critical eigen triples. If  $(\omega_0, \sigma, p)$  is a critical eigen triple,  $\omega_0 > 0$ , then there cannot be another critical eigen triple  $(\omega_1, \sigma, p)$ ,  $\omega_1 > 0$ ,  $\omega_1 \neq \omega_0$ . Furthermore, since  $(-\omega_0, \sigma, p)$  is also a critical eigen triple, there can be no critical eigen triple  $(\omega_2, \sigma, p)$ ,  $\omega_2 < 0$ ,  $\omega_2 \neq -\omega_0$ . This does not preclude two or more lobes crossing. It only refers to a fixed lobe.

Finally we can state the Hopf criteria. That is, there is a family of simple, conjugate eigenvalues  $\lambda(\mu)$ ,  $\bar{\lambda}(\mu)$ , of (6.106), such that

$$\lambda(\mu) = \alpha(\mu) + i\omega_0(\mu), \tag{6.129}$$

where  $\alpha$ ,  $\omega_0$  are real,  $\omega_0(\mu) = \omega(\mu) + \omega_c$  where  $\omega(\mu)$  is a perturbation of a critical frequency  $\omega_c$ , and

$$\begin{aligned}\alpha(0) &= 0, \\ \omega_0(0) &> 0, \\ \alpha'(0) &> 0.\end{aligned}\tag{6.130}$$

The proof of this result depends on the Implicit Function Theorem and is given in Gilsinn [11]. As a consequence of the Implicit Function Theorem the conditions  $\alpha(0) = 0$  and  $\omega(0) = 0$  and thus  $\omega_0(0) > 0$  follows. The last condition, that  $\alpha'(0) > 0$ , follows from following relations, valid for the current machine tool model, that are also shown in Gilsinn [11]

$$\begin{aligned}\alpha'(0) &= \frac{[2\xi - \sigma\omega_c^2 + \sigma(1 + p_c)][1 - \omega_c^2] + [2\omega_c(1 + \sigma\xi)][2\xi\omega_c]}{p_c[2\xi - \sigma\omega_c^2 + \sigma(1 + p_c)]^2 + p_c[2\omega_c(1 + \sigma\xi)]^2}, \\ \omega'(0) &= \frac{[2\xi - \sigma\omega_c^2 + \sigma(1 + p_c)][2\xi\omega_c] - [1 - \omega_c^2][2\omega_c(1 + \sigma\xi)]}{p_c[2\xi - \sigma\omega_c^2 + \sigma(1 + p_c)]^2 + p_c[2\omega_c(1 + \sigma\xi)]^2},\end{aligned}\tag{6.131}$$

The numerator of  $\alpha'(0)$ , divided by  $p_c$ , can be expanded to give

$$2\xi(1 + \omega_c^2) + \sigma(1 - \omega_c^2)^2 + 4\sigma\omega_c^2\xi^2 + \sigma p_c(1 - \omega_c^2).\tag{6.132}$$

The only term in (6.132) that can potentially cause (6.132) to become negative is the last one. However,  $p_c$  and  $\omega_c$  are related by (6.128) which implies that

$$p_c = \frac{(1 - \omega_c^2)^2 + (2\xi\omega_c)^2}{2(\omega_c^2 - 1)}.\tag{6.133}$$

If we substitute (6.133) into (6.132) one gets

$$2\xi(1 + \omega_c^2) + \frac{\sigma}{2}(1 - \omega_c^2)^2 + 2\sigma\omega_c^2\xi^2,\tag{6.134}$$

which is clearly positive so that  $\alpha'(0) > 0$ . To compute the bifurcating periodic solutions and determine their periods later we will need to use both  $\alpha'(0)$  and  $\omega'(0)$ . Finally, the last Hopf condition is also shown in Gilsinn [11] and that is that all other eigenvalues than the two critical ones have negative real parts. The proof of this part of the Hopf result involves a contour integration.

We are now in a position to determine the nature of the Hopf bifurcation that occurs at a critical eigen triple for the machine tool model. To simplify the calculations only the bifurcation at the minimum points of the lobes,  $p_m$  and  $\omega_m$ , given by (6.107) and (6.108), will be examined. Any other point on a lobe would involve more complicated expressions for any  $p$  greater than  $p_m$  and obscure the essential arguments. We will see that a bifurcation, called a *subcritical bifurcation* occurs at this point, which implies that, depending on the initial amplitude used to integrate (6.92), the solution can become unstable in a region that otherwise might be considered a stable region.

The rotation rate,  $\Omega_m$ , at  $p_m$  can be computed from (6.108) and (6.123) as

$$\begin{aligned}\psi_m &= -\pi + \tan^{-1} \left( \sqrt{1 + 2\xi} \right), \\ \Omega_m &= \frac{1}{\sigma_m} = \frac{\omega_m}{2(\psi_m + r\pi) + 3\pi},\end{aligned}\tag{6.135}$$

for  $r = 0, 1, 2, \dots$ . When  $\xi = 0.0135$ , as computed for the current machining model, one has that  $\psi_m = -2.3495$ . When  $r = 0$ ,  $\Omega_m = 0.2144$  ( $\sigma_m = 4.6642$ ), which is the dimensionless rotation rate at the minimum of the first lobe to the right in Fig. 6.4. This point is selected purely in order to illustrate the calculations. The stability limit in Fig. 6.4 is given by (6.107) as

$$p_m = 2\xi(\xi + 1) = 0.027365.\tag{6.136}$$

The frequency at this limit is given by (6.108) as

$$\omega_m = \sqrt{1 + 2\xi} = 1.01341.\tag{6.137}$$

Then from (6.131)

$$\begin{aligned}\alpha'(0) &= \frac{1}{2(1 + \xi)^2(1 + \xi\sigma_m)}, \\ \omega'(0) &= \frac{\sqrt{1 + 2\xi}}{2(1 + \xi)^2(1 + \xi\sigma_m)}.\end{aligned}\tag{6.138}$$

### 6.5.5 Step 5: Apply Orthogonal Decomposition

We will now follow the steps needed to compute the bifurcating periodic solutions on the center manifold. The first step is to compute the eigenvectors for the infinitesimal generators  $A$  and  $A^*$ . The general forms for these eigenvectors are given by (6.58) and (6.59). We wish to compute the constant vectors  $C$  and  $D$ . To compute  $C$  we note that for  $\theta = 0$ , (6.40) implies  $(U + V \exp(-i\omega\sigma))C = i\omega C$  or  $(U - i\omega I + V \exp(-i\omega\sigma))C = 0$ . Since  $i\omega$  is an eigenvalue, (6.56), (6.97), and (6.106) imply that there is a nonzero solution  $C$ . If we set  $c_1 = 1$  it is easy to compute  $c_2 = i\omega$ . The eigenvectors of  $A$  are then given by

$$\begin{aligned}\phi_C(\theta) &= e^{i\omega\theta} \begin{pmatrix} 1 \\ i\omega \end{pmatrix}, \\ \bar{\phi}_C(\theta) &= e^{-i\omega\theta} \begin{pmatrix} 1 \\ -i\omega \end{pmatrix}.\end{aligned}\tag{6.139}$$

To compute  $D$  we have, at  $\theta = 0$ , from (6.47), that  $(U^T + V^T e^{i\omega\sigma} + i\omega I)D = 0$ . The determinant of the matrix on the left is the characteristic equation so that there is a nonzero  $D$ . From (6.58), (6.59), (6.10), and (6.63) one seeks to solve for  $D$  so that

$$\langle \Psi, \Phi \rangle = \begin{pmatrix} \langle \phi_D, \phi_C \rangle & \langle \phi_D, \bar{\phi}_C \rangle \\ \langle \bar{\phi}_D, \phi_C \rangle & \langle \bar{\phi}_D, \bar{\phi}_C \rangle \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \quad (6.140)$$

Due to symmetry one only needs to satisfy  $\langle \phi_D, \phi_C \rangle = 1$  and  $\langle \phi_D, \bar{\phi}_C \rangle = 0$ . From (6.53), (6.58), and (6.59) compute  $d_1$  and  $d_2$  to satisfy

$$\begin{aligned} 1 &= \bar{d}_1 + [\sigma p_c \cos \omega \sigma + i(\omega - \sigma p_c \sin \omega \sigma)] \bar{d}_2, \\ 0 &= \bar{d}_1 + \left[ \frac{p_c}{\omega} \sin \omega \sigma - i\omega \right] \bar{d}_2, \end{aligned} \quad (6.141)$$

from which the eigenvectors of  $A^*$  associated with the eigenvalues  $-i\omega$ ,  $i\omega$  can be computed as

$$\begin{aligned} \phi_D(\theta) &= e^{i\omega\theta} D, \\ \bar{\phi}_D(\theta) &= e^{-i\omega\theta} \bar{D}, \end{aligned} \quad (6.142)$$

where  $D = (d_1, d_2)^T$  and

$$\begin{aligned} d_1 &= -\left( \frac{p_c}{\omega} \sin \omega \sigma + i\omega \right) d_2, \\ d_2 &= \frac{(\sigma p_c \omega^2 \cos \omega \sigma - p_c \omega \sin \omega \sigma) + i(2\omega^3 - \sigma p_c \omega^2 \sin \omega \sigma)}{(\sigma p_c \omega \cos \omega \sigma - p_c \sin \omega \sigma)^2 + (2\omega^2 - \sigma p_c \omega \sin \omega \sigma)^2}, \end{aligned} \quad (6.143)$$

From (6.143) the value of  $\bar{d}_2$  can be calculated as

$$\bar{d}_2 = \frac{-\xi - i\sqrt{1+2\xi}}{2(1+\xi\sigma_c)(1+\xi)^2}, \quad (6.144)$$

where  $\sigma = \sigma_m$ ,  $p_c = p_m$ ,  $\omega = \omega_m$  in (6.143).

Once these eigenvectors have been computed the decomposition then can be written by a straightforward use of (6.98), (6.84), and (6.85).

## 6.6 Computing the Bifurcated Periodic Solution on the Center Manifold

This section will be written in such a manner that it could apply to both ordinary and delay differential equations. We know from ordinary differential equations that in the case of the homogeneous portion having two eigenvalues with zero real parts and all of the others negative real parts there are two manifolds of solutions. One manifold, called the stable manifold, is an invariant manifold of solutions that decay to the equilibrium point. The other manifold, called the center manifold, is an invariant manifold on which the essential behavior of the solution in the neighborhood of the equilibrium point is determined. The step numbers in the next sections should not be confused with the decomposition step numbers. The step numbers in these sections refer to the steps needed to form the center manifold, the normal forms, and the periodic solutions.

### 6.6.1 Step 1: Compute the Center Manifold Form

To begin the construction of a center manifold we start with the equations (6.84)

$$\begin{aligned}\frac{dy}{dt}(t) &= i\omega y(t) + F_1(Y, w_t), \\ \frac{d\bar{y}}{dt}(t) &= -i\omega \bar{y}(t) + \bar{F}_1(Y, w_t), \\ \frac{dw_t}{dt} &= Aw_t + F_2(Y, w_t),\end{aligned}\tag{6.145}$$

where

$$\begin{aligned}F_1(Y, w_t) &= \bar{\phi}_D^T(0)f(z_t), \\ F_2(Y, w_t) &= \begin{cases} -2\text{Re}\{\langle \phi_D, z_t \rangle C\} & -\sigma \leq \theta < 0, \\ -2\text{Re}\{\langle \phi_D, z_t \rangle C\} + f(z_t) & \theta = 0. \end{cases}\end{aligned}\tag{6.146}$$

This system has been decomposed into two equations that have eigenvalues with zero real parts and one that has eigenvalues with negative real parts. For notation, let  $E^c$  be the subspace formed by the eigenvectors  $\phi_C$ ,  $\bar{\phi}_C$ . We can now define a center manifold by a function  $w = w(y, \bar{y})$  for  $|y|$ ,  $|\bar{y}|$  sufficiently small such that  $w(0, 0) = 0$ ,  $\mathcal{D}w(0, 0) = 0$ , where  $\mathcal{D}$  is the total derivative operator. Note that  $w = w(y, \bar{y})$  is only defined locally and the conditions on  $w$  make it tangent to  $E^c$  at  $(0, 0)$ . According to Wiggins [34] the center manifold for (6.84) can then be specified as

$$W^c(0) = \{(y, \bar{y}, w) \in C^3 \mid w = w(y, \bar{y}), |y|, |\bar{y}| < \delta, w(0, 0) = 0, \mathcal{D}w(0, 0) = 0\}.\tag{6.147}$$

for  $\delta$  sufficiently small.

Since the center manifold is invariant, the dynamics of the first two equations in (6.145) must be restricted to the center manifold and satisfy

$$\begin{aligned}\frac{dy}{ds} &= i\omega y + F_1(y, w(y, \bar{y})), \\ \frac{d\bar{y}}{ds} &= -i\omega \bar{y} + \bar{F}_1(y, w(y, \bar{y})).\end{aligned}\tag{6.148}$$

There is also one more condition on the dynamics that must be satisfied by  $w = w(y, \bar{y})$  and that is, it must satisfy the last equation in (6.145). Then, by taking appropriate partial derivatives we must have

$$\begin{aligned}\mathcal{D}_y w(y, \bar{y})\{i\omega y + F_1(y, w(y, \bar{y}))\} + \mathcal{D}_{\bar{y}} w(y, \bar{y})\{-i\omega \bar{y} + \bar{F}_1(y, w(y, \bar{y}))\} \\ = Aw(y, \bar{y}) + F_2(y, w(y, \bar{y})),\end{aligned}\tag{6.149}$$

where  $\mathcal{D}$  represents the derivative with respect to the subscripted variable. The argument of Kazarinoff et al. [22] (see also Carr [9]) can be used to look for a center manifold  $w(y, \bar{y})$  that approximately solves (6.149). We will not need to reduplicate

the argument here but to note that in fact an approximate center manifold, satisfying (6.149), can be given as a quadratic form in  $y$  and  $\bar{y}$  with coefficients as functions of  $\theta$

$$w(y, \bar{y})(\theta) = w_{20}(\theta)\frac{y^2}{2} + w_{11}(\theta)y\bar{y} + w_{02}(\theta)\frac{\bar{y}^2}{2}. \quad (6.150)$$

Then the projected equation (6.148) on the center manifold takes the form

$$\begin{aligned} \frac{dy}{ds} &= i\omega y + g(y, \bar{y}), \\ \frac{d\bar{y}}{ds} &= -i\omega\bar{y} + \bar{g}(y, \bar{y}), \end{aligned} \quad (6.151)$$

where the  $g(y, \bar{y})$  is given by

$$g(y, \bar{y}) = g_{20}\frac{y^2}{2} + g_{11}y\bar{y} + g_{02}\frac{\bar{y}^2}{2} + g_{21}\frac{y^2\bar{y}}{2}. \quad (6.152)$$

Since  $w_{02} = \bar{w}_{20}$ , one only needs to solve for  $w_{20}$  and  $w_{11}$ . It can be shown that  $w_{20}$ ,  $w_{11}$  take the form

$$\begin{aligned} w_{20}(\theta) &= c_1\phi(\theta) + c_2\bar{\phi}(\theta) + M e^{2i\omega\theta}, \\ w_{11}(\theta) &= c_3\phi(\theta) + c_4\bar{\phi}(\theta) + N, \end{aligned} \quad (6.153)$$

where  $c_i, i = 1, \dots, 4$  are constants and  $M, N$  are vectors. We will show how the coefficients and the vectors can be computed in a specific example below.

### 6.6.2 Step 2: Develop the Normal Form on the Center Manifold

The equations on the center manifold can be further simplified. Normal form theory, following the argument of Wiggins [34] (see also Nayfeh [25]), can be used to reduce (6.151) to the simpler form (6.154) below on the center manifold. In fact, (6.148) is reduced to a normal form by a transformation of variables,  $y \rightarrow v$ , so that the new system takes the form

$$\begin{aligned} \frac{dv}{ds} &= i\omega v + c_{21}v^2\bar{v}, \\ \frac{d\bar{v}}{ds} &= -i\omega\bar{v} + c_{21}\bar{v}^2v, \end{aligned} \quad (6.154)$$

where the higher order terms have been dropped. The derivation of this formula is complex and is not needed for this chapter. The interested reader should consult Gilsinn [11] and Wiggins [34] for the essential ideas involved. The formula needed is one that links (6.154) with (6.152) and is given by

$$c_{21} = \frac{i}{2\omega} \left\{ g_{11}g_{20} - 2|g_{11}|^2 - \frac{|g_{02}|^2}{3} \right\} + \frac{g_{21}}{2}. \quad (6.155)$$

For a more general discussion of the normal form on the center manifold when  $\mu \neq 0$  see Hassard et al. [17]. Up to this point one has only needed  $\mu = 0$ . But, to compute the periodic solution we will reintroduce  $\mu \neq 0$ . A formula for the periodic solutions of (6.1) can be computed. The argument is based on that of Hassard et al. [17] and will not be given but the references Gilsinn [11] and Hassard et al. [17] can be consulted by the interested reader. What is important here is a formula that can be used in specific examples.

### 6.6.3 Step 3: Form the Periodic Solution on the Center Manifold

To begin with let  $\varepsilon > 0$  and an initial condition for a periodic solution of (6.154) be given as

$$v(0; \varepsilon) = \varepsilon. \quad (6.156)$$

Then, there exists a family of periodic solutions  $v(s, \mu(\varepsilon))$  of (6.194) with

$$\begin{aligned} \mu(\varepsilon) &= \mu_2 \varepsilon^2 + \cdots, \\ \beta(\varepsilon) &= \beta_2 \varepsilon^2 + \cdots, \\ T(\varepsilon) &= T_0(1 + \tau_2 \varepsilon^2 + \cdots), \end{aligned} \quad (6.157)$$

where  $T(\varepsilon)$  is the period of  $v(s, \mu(\varepsilon))$ ,  $\beta(\varepsilon)$  is the nonzero characteristic exponent, and

$$\begin{aligned} \mu_2 &= -\frac{\operatorname{Re}\{c_{21}\}}{\alpha'(0)}, \\ \beta_2 &= 2\operatorname{Re}\{c_{21}(0)\}, \\ \tau_2 &= -\frac{1}{\omega} (\mu_2 \omega'(0) + \operatorname{Im}\{c_{21}\}), \\ T_0 &= \frac{2\pi}{\omega}. \end{aligned} \quad (6.158)$$

Furthermore,  $v(s, \mu(\varepsilon))$  can be transformed into a family of periodic solutions for (6.1) given by

$$z(s) = \mathcal{P}(s, \mu(\varepsilon)) = 2\varepsilon \operatorname{Re}\left\{\phi(0)e^{i\omega s}\right\} + \varepsilon^2 \operatorname{Re}\left\{M e^{2i\omega s} + N\right\}. \quad (6.159)$$

with  $\varepsilon = (\mu/\mu_2)^{1/2}$ . For  $\mu_2 > 0$  the Hopf bifurcation is called supercritical and for  $\mu_2 < 0$  it is called subcritical. Finally, note that since  $\mu \approx \mu_2 \varepsilon^2$  one can take  $\varepsilon = (\mu/\mu_2)^{1/2}$  which allows one to associate  $Z(s)$  with the parameter  $p = \mu + p_c$ . From Floquet theory if  $\beta(\varepsilon) < 0$  the periodic solution is stable and if  $\beta(\varepsilon) > 0$  it is unstable.

## 6.7 A Machine Tool DDE Example: Part 2

In this section, we will show how the formulas in Sect. 6.6 are computed for the specific example of the turning center model. We will conclude this section with a formula approximating the periodic solution of (6.96).

### 6.7.1 Step 1: Compute the Center Manifold Form

With the eigenvectors computed we can proceed to approximate the center manifold and the projected equations on the center manifold. We begin by assuming that an approximate center manifold, satisfying (6.149), can be given as a quadratic form in  $y$  and  $\bar{y}$  with coefficients as functions of  $\theta$

$$w(y, \bar{y})(\theta) = w_{20}(\theta) \frac{y^2}{2} + w_{11}(\theta) y\bar{y} + w_{02}(\theta) \frac{\bar{y}^2}{2}. \quad (6.160)$$

The object of this section is to compute the constants  $c_1, c_2, c_3, c_4$ , and the vectors  $M, N$  in (6.153) in order to create the coefficients for (6.150).

Using (6.70) we can introduce coordinates on the center manifold by

$$z = w + \phi y + \bar{\phi} \bar{y}. \quad (6.161)$$

From (6.83), (6.84), (6.148), and (6.161) we have on the center manifold

$$\frac{dy}{ds} = i\omega y + \bar{\phi}^{*T}(0) f(w(y, \bar{y}) + \phi y + \bar{\phi} \bar{y}). \quad (6.162)$$

Define

$$g(y, \bar{y}) = \bar{\phi}^{*T}(0) f(w(y, \bar{y}) + \phi y + \bar{\phi} \bar{y}), \quad (6.163)$$

where  $\bar{\phi}^{*T}(0) = (\bar{d}_1, \bar{d}_2)$  from (6.142) and

$$f(w(y, \bar{y}) + \phi y + \bar{\phi} \bar{y}) = \begin{pmatrix} 0 \\ p_c \left( E \left[ w(y, \bar{y})_1(-\sigma) + y\phi_1(-\sigma) + \bar{y}\bar{\phi}_1(\sigma) - w(y, \bar{y})_1(0) - y\phi_1(0) - \bar{y}\bar{\phi}_1(0) \right]^2 \right. \\ \left. + E \left[ w(y, \bar{y})_1(-\sigma) + y\phi_1(-\sigma) + \bar{y}\bar{\phi}_1(\sigma) - w(y, \bar{y})_1(0) - y\phi_1(0) - \bar{y}\bar{\phi}_1(0) \right]^3 \right) \end{pmatrix}. \quad (6.164)$$

From (6.150)

$$\begin{aligned} w(y, \bar{y})(0) &= w_{20}(0) \frac{y^2}{2} + w_{11}(0) y\bar{y} + w_{02}(0) \frac{\bar{y}^2}{2}, \\ w(y, \bar{y})(-\sigma) &= w_{20}(-\sigma) \frac{y^2}{2} + w_{11}(-\sigma) y\bar{y} + w_{02}(-\sigma) \frac{\bar{y}^2}{2}, \end{aligned} \quad (6.165)$$

where  $w_{ij}(\theta) = \left( w_{ij}^1(\theta), w_{ij}^2(\theta) \right)^T$ .

Note here that in order to compute  $\mu_2$ ,  $\tau_2$ ,  $\beta_2$  one need only determine  $g(y, \bar{y})$  in the form (6.152). To find the coefficients for (6.152) begin by expanding the nonlinear terms of (6.164) up to cubic order, keeping only the cubic term  $y^2\bar{y}$ . To help simplify the notation let

$$\gamma = e^{-i\omega\sigma} - 1. \quad (6.166)$$

Then, using (6.58), (6.165), and (6.166),

$$\begin{aligned} & E [w(y, \bar{y})_1(-\sigma) + y\phi_1(-\sigma) + \bar{y}\bar{\phi}_1(\sigma) - w(y, \bar{y})_1(0) - y\phi_1(0) - \bar{y}\bar{\phi}_1(0)]^2 \\ &= Ey^2\gamma^2 + 2Ey\bar{y}\gamma\bar{\gamma} \\ &+ E\bar{y}^2\bar{\gamma}^2 + E \{ [w_{20}^1(-\sigma) - w_{20}^1(0)] \bar{\gamma} + 2 [w_{11}^1(-\sigma) - w_{11}^1(0)] \gamma \} y^2\bar{y}, \quad (6.167) \\ & E [w(y, \bar{y})_1(-\sigma) + y\phi_1(-\sigma) + \bar{y}\bar{\phi}_1(\sigma) - w(y, \bar{y})_1(0) - y\phi_1(0) - \bar{y}\bar{\phi}_1(0)]^3 \\ &= 3E\gamma^2\bar{\gamma}y^2\bar{y}. \end{aligned}$$

If we define

$$\begin{aligned} g_{20} &= 2E\gamma^2\bar{d}_2p_c, \\ g_{11} &= 2E\gamma\bar{\gamma}\bar{d}_2p_c, \\ g_{02} &= 2E\bar{\gamma}^2\bar{d}_2p_c, \end{aligned} \quad (6.168)$$

and

$$g_{21} = 2p_c \{ E [w_{20}^1(-\sigma) - w_{20}^1(0)] \bar{\gamma} + 2E [w_{11}^1(-\sigma) - w_{11}^1(0)] \gamma + 3E\gamma^2\bar{\gamma} \} \bar{d}_2, \quad (6.169)$$

we can use (6.163)–(6.169) to write

$$f(w(y, \bar{y}) + \phi y + \bar{\phi}\bar{y}) = \left\{ \frac{g_{20}y^2}{2\bar{d}_2} + \frac{g_{11}y\bar{y}}{\bar{d}_2} + \frac{g_{02}\bar{y}^2}{2\bar{d}_2} + \frac{g_{21}y^2\bar{y}}{2\bar{d}_2} \right\} \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \quad (6.170)$$

In order to complete the computation of  $g_{21}$  one needs to compute the center manifold coefficients  $w_{20}$ ,  $w_{11}$ .

Since we are looking for the center manifold as a quadratic form, we need only expand functions in terms of  $y^2$ ,  $y\bar{y}$ ,  $\bar{y}^2$ . From the definition of  $F_2$  in (6.85), (6.152), and (6.163) write  $F_2$  as

$$\begin{aligned} F_2(y, \bar{y})(\theta) &= - \{ g_{20}\phi(\theta) + \bar{g}_{02}\bar{\phi}(\theta) \} \frac{y^2}{2} \\ &\quad - \{ g_{11}\phi(\theta) + \bar{g}_{11}\bar{\phi}(\theta) \} y\bar{y} \\ &\quad - \{ g_{02}\phi(\theta) + \bar{g}_{20}\bar{\phi}(\theta) \} \frac{\bar{y}^2}{2}, \end{aligned} \quad (6.171)$$

for  $-\sigma \leq \theta < 0$  and for  $\theta = 0$

$$\begin{aligned}
F_2(y, \bar{y})(0) = & - \left\{ g_{20}\phi(0) + \bar{g}_{02}\bar{\phi}(0) - \frac{g_{20}}{d_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} \frac{y^2}{2} \\
& - \left\{ g_{11}\phi(0) + \bar{g}_{11}\bar{\phi}(0) - \frac{g_{11}}{d_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} y\bar{y} \\
& - \left\{ g_{02}\phi(0) + \bar{g}_{20}\bar{\phi}(0) - \frac{g_{02}}{d_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} \frac{\bar{y}^2}{2}.
\end{aligned} \tag{6.172}$$

Note that, to compute the coefficients of the center manifold, one only needs to work to the second order.

Since  $g_{02}/\bar{d}_2 = \bar{g}_{20}/d_2$  write the coefficients of  $F_2(y, \bar{y})$  as

$$\begin{aligned}
F_{20}^2(\theta) &= \begin{cases} - (g_{20}\phi(\theta) + \bar{g}_{02}\bar{\phi}(\theta)) & -\sigma \leq \theta < 0, \\ - \left( g_{20}\phi(0) + \bar{g}_{02}\bar{\phi}(0) - \frac{g_{20}}{d_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) & \theta = 0, \end{cases} \\
F_{11}^2(\theta) &= \begin{cases} - (g_{11}\phi(\theta) + \bar{g}_{11}\bar{\phi}(\theta)) & -\sigma \leq \theta < 0, \\ - \left( g_{11}\phi(0) + \bar{g}_{11}\bar{\phi}(0) - \frac{g_{11}}{d_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) & \theta = 0, \end{cases} \\
F_{02}^2(\theta) &= \bar{F}_{20}^2(\theta).
\end{aligned} \tag{6.173}$$

One can now set up equation (6.149) to approximate the center manifold. On this manifold one must have

$$W(s) = w(y(s), \bar{y}(s)). \tag{6.174}$$

By taking derivatives, the equation for the manifold becomes

$$w_y(y, \bar{y})y'(s) + w_{\bar{y}}(y, \bar{y})\bar{y}'(s) = Aw(y(s), \bar{y}(s)) + F_2(y(s), \bar{y}(s)), \tag{6.175}$$

where  $w(y, \bar{y})$  is given by (6.150). The partial derivatives are given by

$$\begin{aligned}
w_y(y, \bar{y}) &= w_{20}y + w_{11}\bar{y}, \\
w_{\bar{y}}(y, \bar{y}) &= w_{11}y + w_{02}\bar{y}.
\end{aligned} \tag{6.176}$$

Using (6.151) and (6.176) expand the terms of (6.175) to second order as

$$\begin{aligned}
w_y(y, \bar{y})(\theta)y'(s) &= i\omega w_{20}(\theta)y^2(s) + i\omega w_{11}(\theta)\bar{y}(s)y(s), \\
w_{\bar{y}}(y, \bar{y})(\theta)\bar{y}'(s) &= -i\omega w_{11}(\theta)y(s)\bar{y}(s) - i\omega w_{02}(\theta)\bar{y}^2(s), \\
Aw(y, \bar{y})(\theta) &= (Aw_{20})(\theta)\frac{y^2}{2} + (Aw_{11})(\theta)y\bar{y} + (Aw_{02})(\theta)\frac{\bar{y}^2}{2}, \\
F_2(y, \bar{y})(\theta) &= F_{20}^2(\theta)\frac{y^2}{2} + F_{11}^2(\theta)y\bar{y} + F_{02}^2(\theta)\frac{\bar{y}^2}{2}.
\end{aligned} \tag{6.177}$$

Substitute (6.177) into (6.175) and equate coefficients to get

$$\begin{aligned}
2i\omega w_{20}(\theta) - Aw_{20} &= F_{20}^2(\theta) \\
-Aw_{11} &= F_{11}^2(\theta), \\
-2i\omega w_{02}(\theta) - Aw_{02} &= F_{02}^2(\theta).
\end{aligned} \tag{6.178}$$

Since  $F_{02}^2 = \bar{F}_{20}^2$  and  $w_{02} = \bar{w}_{20}$ , one only needs to solve for  $w_{20}$  and  $w_{11}$ .

To compute  $c_3$ ,  $c_4$ ,  $N$ , use the second equation in (6.178), the definition of  $A$  in (6.40) and (6.173). Then for  $-\sigma \leq \theta < 0$

$$\frac{dw_{11}}{d\theta}(\theta) = g_{11}\phi(\theta) + \bar{g}_{11}\bar{\phi}(\theta). \tag{6.179}$$

Integrate (6.179) and use (6.58) to get

$$w_{11}(\theta) = \frac{g_{11}}{i\omega}\phi(\theta) - \frac{\bar{g}_{11}}{i\omega}\bar{\phi}(\theta) + N. \tag{6.180}$$

Clearly

$$\begin{aligned}
c_3 &= \frac{g_{11}}{i\omega}, \\
c_4 &= -\frac{\bar{g}_{11}}{i\omega}.
\end{aligned} \tag{6.181}$$

To determine  $N$  we will use (6.40) for  $\theta = 0$ ,  $\mu = 0$  and the fact that  $\phi(0)$ ,  $\bar{\phi}(0)$  are eigenvectors of  $A$  with eigenvalues  $i\omega$ ,  $-i\omega$  at  $\theta = 0$ . The eigenvector property implies

$$\begin{aligned}
U\phi(0) + V\phi(-\sigma) &= i\omega\phi(0), \\
U\bar{\phi}(0) + V\bar{\phi}(-\sigma) &= -i\omega\bar{\phi}(0).
\end{aligned} \tag{6.182}$$

If we combine this with (6.173), then it is straightforward to show that

$$(U + V)N = -\frac{g_{11}}{d_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tag{6.183}$$

which can be solved for

$$N = -\frac{g_{11}}{d_2} \begin{pmatrix} -1 \\ 0 \end{pmatrix}. \tag{6.184}$$

To solve for  $c_1$ ,  $c_2$ ,  $M$ , use the definition of  $A$  in (6.40) for  $-\sigma \leq \theta < 0$ , (6.173), (6.178) to get

$$\frac{dw_{20}}{d\theta} = 2i\omega w_{20}(\theta) + g_{20}\phi(\theta) + \bar{g}_{02}\bar{\phi}(\theta). \tag{6.185}$$

This nonhomogeneous system has the solution

$$w_{20}(\theta) = -\frac{g_{20}}{i\omega}\phi(\theta) - \frac{\bar{g}_{02}}{3i\omega}\bar{\phi}(\theta) + Me^{2i\omega\theta}. \tag{6.186}$$

Again, clearly

$$\begin{aligned} c_1 &= -\frac{g_{20}}{i\omega}, \\ c_2 &= -\frac{\bar{g}_{02}}{3i\omega}. \end{aligned} \quad (6.187)$$

To solve for  $M$  use the definition of (6.40) for  $\theta = 0$ ,  $\mu = 0$ , (6.173), and again the fact that  $\phi(0)$ ,  $\bar{\phi}(0)$  are eigenvectors of  $A$  with eigenvalues  $i\omega$ ,  $-i\omega$  at  $\theta = 0$  to show that

$$\left(2i\omega I - U - Ve^{-2i\omega\sigma}\right)M = \frac{g_{20}}{\bar{d}_2} \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad (6.188)$$

which can be solved for  $M$  as

$$M = \frac{g_{20}}{\bar{d}_2\Delta} \begin{pmatrix} 1 \\ 2i\omega \end{pmatrix}, \quad (6.189)$$

where

$$\Delta = -4\omega^2 + 4i\xi\omega + p_c \left(1 - e^{2i\omega\sigma}\right). \quad (6.190)$$

One can now return to (6.169) and use (6.180)–(6.184) and (6.186)–(6.190) to construct  $g_{21}$  in (6.191), which concludes the construction of (6.152) and thus the projected equation (6.151) on the center manifold. Then the projected equation (6.148) on the center manifold takes the form (6.151) and (6.152). The coefficients  $g_{ij}$  for (6.152) are given by

$$\begin{aligned} g_{20} &= 2E\gamma^2\bar{d}_2p_c, \\ g_{11} &= 2E\gamma\bar{\gamma}\bar{d}_2p_c, \\ g_{02} &= 2E\bar{\gamma}^2\bar{d}_2p_c, \\ g_{21} &= 2p_c \left\{ E \left[ -\frac{g_{20}\gamma}{i\omega} - \frac{\bar{g}_{02}}{3i\omega} + \frac{g_{20}(e^{-2i\omega\sigma} - 1)}{\bar{d}_2\Delta} \right] \bar{\gamma} \right. \\ &\quad \left. + 2E \left[ \frac{g_{11}\gamma}{i\omega} - \frac{\bar{g}_{11}\bar{\gamma}}{i\omega} \right] \gamma + 3E\gamma^2\bar{\gamma} \right\} \bar{d}_2, \end{aligned} \quad (6.191)$$

where

$$\begin{aligned} \gamma &= e^{-i\omega\sigma} - 1, \\ \Delta &= -4\omega^2 + 4i\xi\omega + p_c \left(1 - e^{2i\omega\sigma}\right). \end{aligned} \quad (6.192)$$

### 6.7.2 Step 2: Develop the Normal Form on the Center Manifold

Once the constants (6.191) have been developed then the normal form on the center manifold is computed as (6.154) using (6.155). Normal form theory, following the

argument of Wiggins [34] (see also Nayfeh [25]), can be used to reduce (6.151) to the simpler form (6.154) on the center manifold. In fact, the system (6.151) can be reduced by a near identity transformation to (6.154) where

$$c_{21} = \frac{i}{2\omega} \left\{ g_{11}g_{20} - 2|g_{11}|^2 - \frac{|g_{02}|^2}{3} \right\} + \frac{g_{21}}{2}. \quad (6.193)$$

The proof of the result is detailed and the reader is referred to Gilsinn [11] and Wiggins [34]. As shown in Hassard et al. [17] the general normal form for the case  $\mu \neq 0$  is given by

$$\frac{dv}{ds} = \lambda(\mu)v + c_{21}(\mu)v^2\bar{v} \quad (6.194)$$

where  $\lambda(0) = i\omega$  and  $c_{21}(0)$  is given by (6.155).

One can now compute  $g_{20}$ ,  $g_{11}$ ,  $g_{02}$ , and  $g_{21}$  from (6.191). Then from (6.155) one computes  $c_{21}$  and finally, from (6.158), one can compute  $\mu_2$ ,  $\tau_2$ ,  $\beta_2$  as

$$\begin{aligned} \mu_2 &= -0.09244, \\ \tau_2 &= 0.002330, \\ \beta_2 &= 0.08466. \end{aligned} \quad (6.195)$$

This implies that at the lobe boundary the DDE bifurcates into a family of unstable periodic solutions in a subcritical manner.

### 6.7.3 Step 3: Form the Periodic Solution on the Center Manifold

Using (6.159), one can compute the form of the bifurcating solutions for (6.96) as

$$z(s) = \begin{pmatrix} 2\varepsilon \cos 1.01341s + \varepsilon^2 ((-2.5968e - 6) \cos 2.02682s \\ -0.014632 \sin 2.02682s + 0.060113) \\ -2\varepsilon \sin 1.01341s + \varepsilon^2 (-0.02966 \cos 2.02682s \\ + (5.2632e - 6) \sin 2.02682s) \end{pmatrix}. \quad (6.196)$$

As noted at the end of Sect. 6.6 one can take as an approximation

$$\varepsilon = \left( \frac{\mu}{\mu_2} \right)^{1/2}. \quad (6.197)$$

It is clear from (6.195) that  $\mu$  must be negative. Thus select

$$\varepsilon = \frac{(-\mu)^{1/2}}{0.3040395}. \quad (6.198)$$

The period of the solution can be computed as

$$T(\varepsilon) = \frac{2\pi}{\omega_m} (1 + \tau_2 \varepsilon^2) = 6.2000421 (1 + 0.002330 \varepsilon^2), \quad (6.199)$$

and the characteristic exponent is given by

$$\beta = 0.08466 \varepsilon^2. \quad (6.200)$$

## 6.8 Simulation Results

To compare with the theoretical results, simulations were performed by direct integration of (6.96). The first of two sets of simulations was initialized at the points A through F in Fig. 6.5 along the vertical line  $\Omega = 0.2144$  (selected for ease of calculation only). This line crosses the minimum of the first lobe in Fig. 6.5. The simulations numerically demonstrate that there are three branches of periodic solutions emanating from the critical bifurcation point  $p_m = 0.027365$ . The three branches are shown in Fig. 6.6. The amplitudes in this figure were computed using (6.196) with  $\varepsilon$  given by (6.197). Two of the branches are unstable and one is stable in the following sense. Solutions initialized below the subcritical branch converge to the zero solution. Those initialized above the subcritical branch grow in amplitude. The solutions initialized above zero for bifurcation parameter values greater than the critical value grow in amplitude. Similar results would be obtained along the lines crossing at other critical points on the lobes.

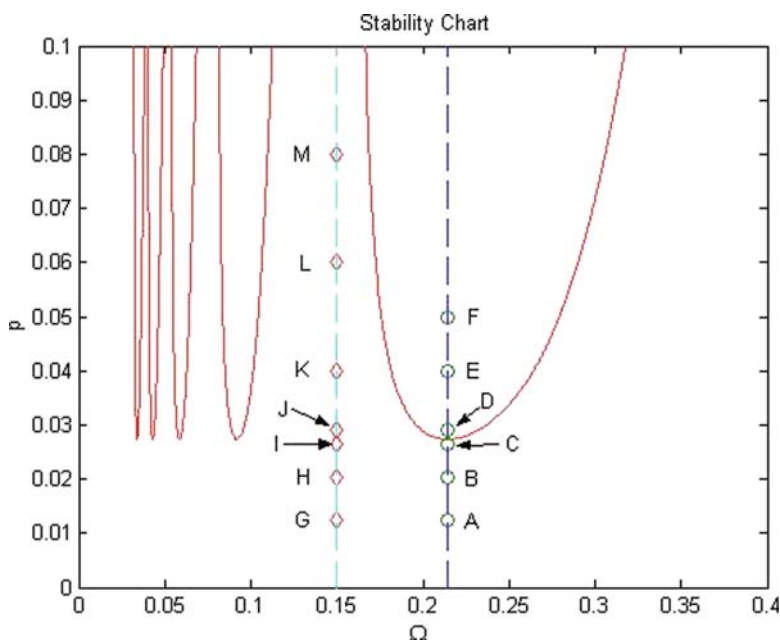
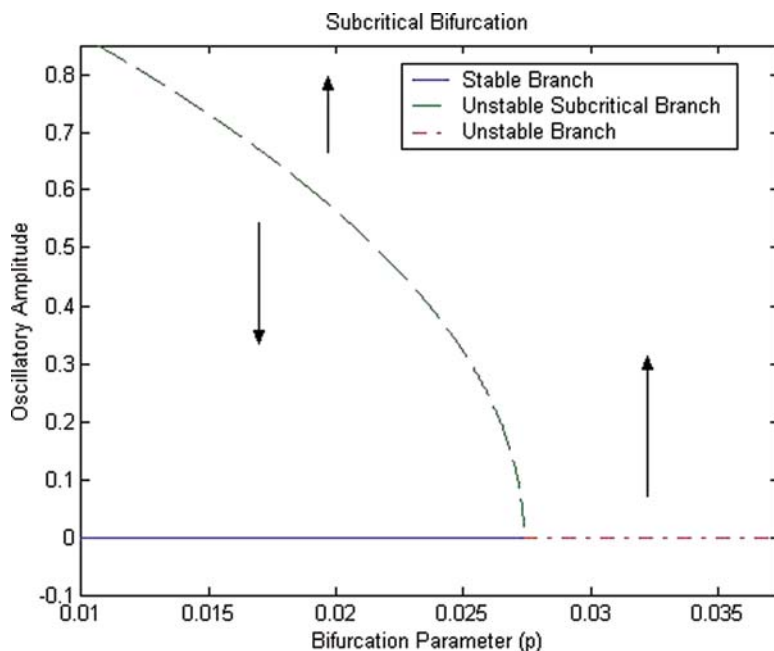


Fig. 6.5 Locations of sample simulated solutions



**Fig. 6.6** Amplitudes of bifurcating branches of solutions at subcritical point

The Hopf bifurcation result is very local around the boundary and only for very small initial amplitudes is it possible to track the unstable limit cycles along the branching amplitude curve. This is shown in Fig. 6.7 where initial simulation locations were selected along the subcritical curve and the DDE was integrated forward over five delay intervals. Note that nearer the critical bifurcation point the solution amplitude remains near the initial value, whereas further along the curve the solution amplitude drops away significantly from the subcritical curve.

Since the subcritical bifurcation curve in Fig. 6.6 is itself an approximation to the true subcritical curve, solutions initialized on the curve tend to decay to zero. This occurs at points A, B, and C in Fig. 6.5.

The decay at point B, when initialized on the subcritical curve is similar to the result at point A, but with a less rapid decay (Fig. 6.8). However, one can show the effect of initializing a solution above the curve at point B. That is shown in Fig. 6.9. Point B is given by  $(\Omega, p) = (0.2144, 0.020365)$ . For  $p = 0.020365$  the subcritical curve amplitude value is 0.550363 (Fig. 6.10). Figures 6.11 and 6.12 show the solution growing when the simulation amplitude is initialized at 0.8.

At point C, when the solution is initialized on the subcritical bifurcation curve, the phase plot remains very close to a periodic orbit, indicating that the Hopf results are very local in being able to predict the unstable periodic solution.

The behavior at points D, E, and F of Fig. 6.5 are similar in that all of the solutions initialized above zero experience growth and eventually explode numerically.

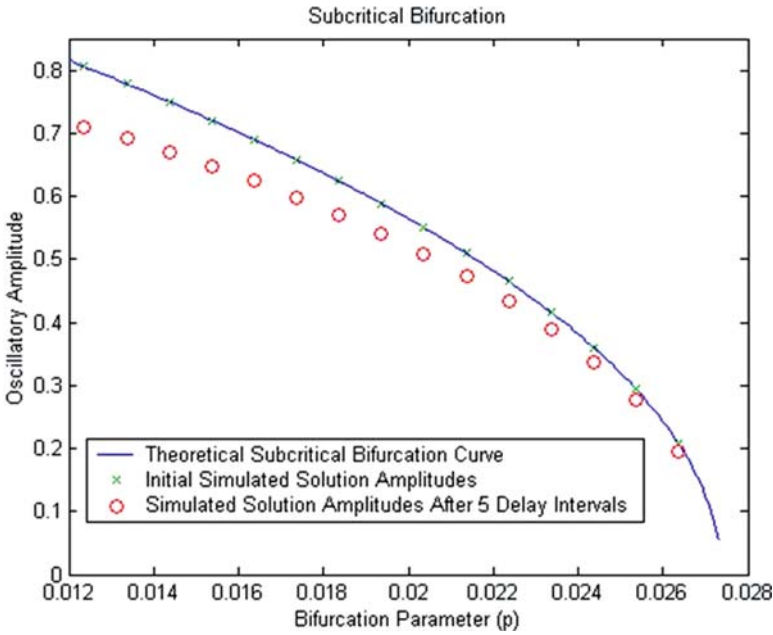


Fig. 6.7 Theoretical and simulated subcritical solution amplitudes

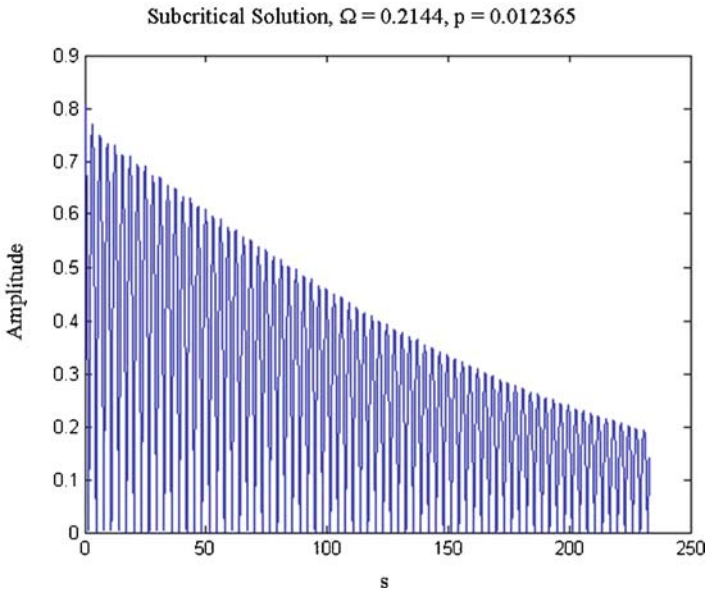
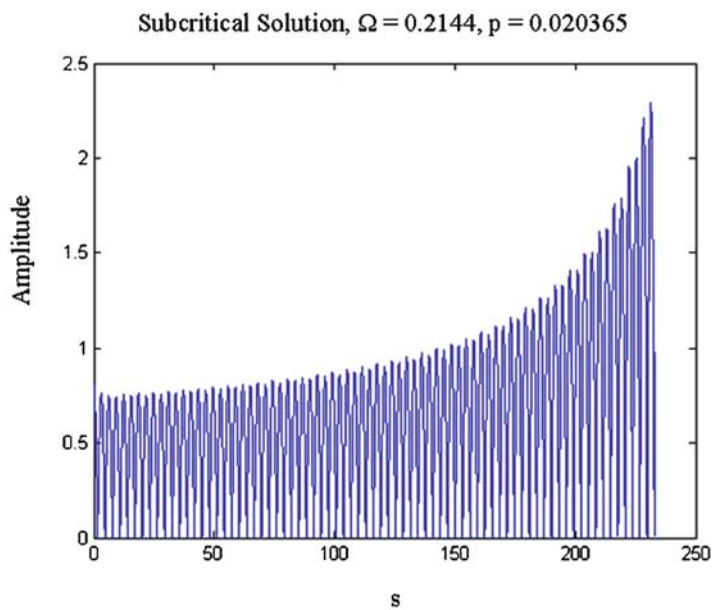
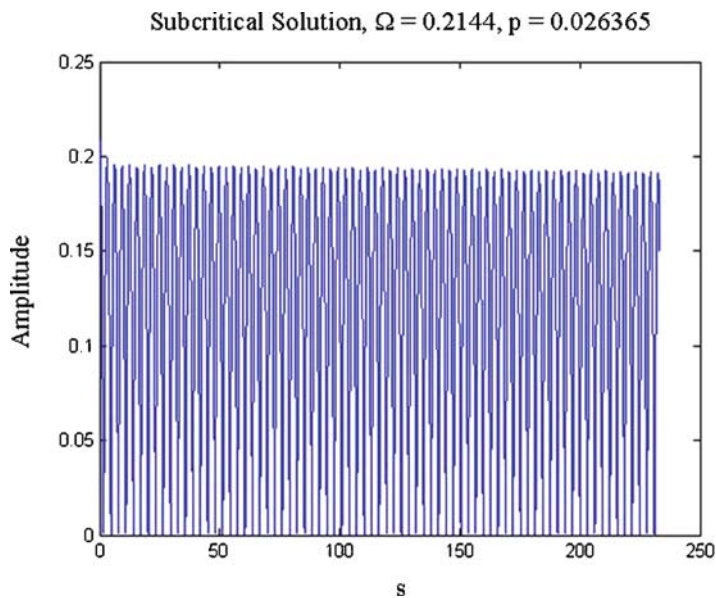


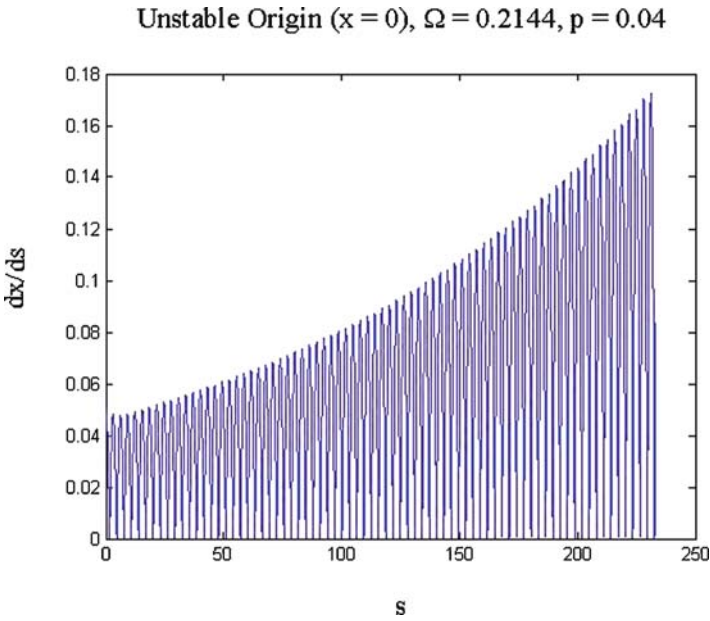
Fig. 6.8 Stable amplitude of solution at point A



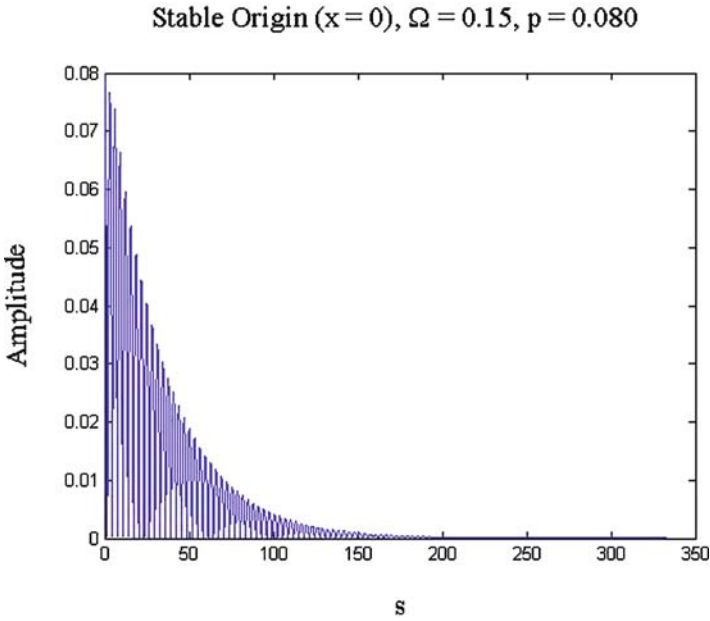
**Fig. 6.9** Unstable amplitude of solution at point B



**Fig. 6.10** Nearly stable amplitude of solution at point C



**Fig. 6.11** Unstable amplitude of solution at point E



**Fig. 6.12** Stable amplitude of solution at point M

The second set of simulations, initialized at points G through M along the line  $\Omega = 0.15$ , in Fig. 6.5, shows the stability of solutions for parameters falling between lobes in that the solutions of the DDE (6.92) all decay to zero. The gaps between lobes are significant for machining. Since the parameter  $p$  is proportional to chip width, the larger the  $p$  value for which the system is stable the more material can be removed without chatter, where chatter can destroy the surface finish of the workpiece. These large gaps tend to appear between the lobes in high-speed machining with spindle rotation rates of the order of  $2,094.4 \text{ rad s}^{-1}$  (20,000 rpm) or greater. Figure 6.12 illustrates this stability with one time plot for point M,  $(\Omega, p) = (0.15, 0.08)$ . The solution for this plot is initialized at amplitude 0.08.

**Acknowledgments** The author wishes to thank Dr. Tim Burns of NIST and Prof. Dianne O’Leary of the University of Maryland for their careful reading and gracious assistance in preparing this chapter.

## References

- Altintas, Y. and Budak, E., Analytic prediction of stability lobes in milling, *Ann. CIRP* **44**, 1995, 357–362.
- an der Heiden, U. Delays in physiological systems, *J. Math. Biol.* **8**, 1979, 345–364.
- Arnold, V.I., *Ordinary Differential Equations*, Springer, Berlin, 1992.
- Avellar, C.E. and Hale, J.K., On the zeros of exponential polynomials, *J. Math. Anal. Appl.* **73**, 1980, 434–452.
- Balachandran, B., Nonlinear dynamics of milling processes, *Philos. Trans. R. Soc. Lond. A* **359**, 2001, 793–819.
- Balachandran, B. and Zhao, M.X., A mechanics based model for study of dynamics of milling operations, *Meccanica* **35**, 2000, 89–109.
- Bellman, R. and Cooke, K.L., *Differential–Difference Equations*, Academic, New York, NY, 1963.
- Campbell, S.A., Calculating centre manifolds for delay differential equations using maple, In: Balachandran, B., Gilsinn, D.E., Kalmár-Nagy, T. (eds), *Delay Differential Equations: Recent Advances and New Directions*, Springer, New York, NY, p. TBD
- Carr, J., *Applications of Centre Manifold Theory*, Springer, New York, NY, 1981.
- Cronin, J., *Differential Equations: Introduction and Qualitative Theory*, Marcel Dekker, New York, NY, 1980.
- Gilsinn, D.E., Estimating critical Hopf bifurcation parameters for a second-order differential equation with application to machine tool chatter, *Nonlinear Dynam.* **30**, 2002, 103–154.
- Hale, J.K., Linear functional-differential equations with constant coefficients, *Contrib. Diff. Eqns.* **II**, 1963, 291–317.
- Hale, J.K., *Ordinary Differential Equations*, Wiley, New York, NY, 1969.
- Hale, J., *Functional Differential Equations*, Springer, New York, NY, 1971.
- Hale, J.K. and Lunel, S.M.V., *Introduction to Functional Differential Equations*, Springer, New York, NY, 1993.
- Hassard, B. and Wan, Y.H., Bifurcation formulae derived from center manifold theory, *J. Math. Anal. Appl.* **63**, 1978, 297–312.
- Hassard, B.D., Kazarinoff, N.D., and Wan, Y.H., *Theory and Applications of Hopf Bifurcations*, Cambridge University Press, Cambridge, 1981.
- Hille, E. and Phillips, R. S., *Functional Analysis and Semi-Groups*, American Mathematical Society, Providence, RI, 1957.

19. Hirsch, M.W. and Smale, S., *Differential Equations, Dynamical Systems, and Linear Algebra*, Academic, New York, NY, 1974.
20. Kalmár-Nagy, T., Pratt, J.R., Davies, M.A., and Kennedy, M.D., Experimental and analytical investigation of the subcritical instability in metal cutting, *Proceedings of DETC'99*, 17th ASME Biennial Conference on Mechanical Vibration and Noise, Las Vegas, Nevada, Sept. 12–15, 1999, 1–9.
21. Kalmár-Nagy, T., Stépán G., Moon, F.C., Subcritical Hopf bifurcation in the delay equation model for machine tool vibrations. *Nonlinear Dynam.* 26, 2001, 121–142
22. Kazarinoff, N.D., Wan, Y.-H., and van den Driessche, P., Hopf bifurcation and stability of periodic solutions of differential-difference and integro-differential equations', *J. Inst. Math. Appl.* 21, 1978, 461–477.
23. Kuang, Y., *Delay Differential Equations with Applications in Population Dynamics*, Academic, Boston, MA, 1993.
24. MacDonald, N. *Biological Delay Systems: Linear Stability Theory*, Cambridge University Press, Cambridge, 1989.
25. Nayfeh, A.H., *Method of Normal Forms*, Wiley, New York, NY, 1993.
26. Nayfeh, A.H. and Balachandran, B. *Applied Nonlinear Dynamics*, Wiley, New York, NY, 1995.
27. Pinney, E., *Ordinary Difference–Differential Equations*, University of California Press, Berkeley, CA, 1958.
28. Stépán, G., *Retarded Dynamical Systems: Stability and Characteristic Functions*, Longman Scientific & Technical, Harlow, England, 1989.
29. Stone, E. and Askari A., Nonlinear models of chatter in drilling processes. *Dynam. Syst.* 17(1), 2002, 65–85.
30. Stone E, Campbell SA (2004) Stability and bifurcation analysis of a nonlinear DDE model for drilling. *Journal of Nonlinear Science* 14 (1), 27–57.
31. Taylor, J.R., 'On the art of cutting metals', *American Society of Mechanical Engineers*, New York, 1906.
32. Tlustý, J., 'Machine Dynamics', *Handbook of High-speed Machine Technology*, Robert I. King, Editor, Chapman and Hall, New York, 1985, 48–153.
33. Tobias, S.A. and Fishwick, W., 'The Chatter of Lathe Tools Under Orthogonal Cutting Conditions', *Transactions of the ASME* 80, 1958, 1079–1088.
34. Wiggins, S., *Introduction to Applied Nonlinear Dynamical Systems and Chaos*, Springer-Verlag, New York, 1990.
35. Yosida, K., *Functional Analysis*, Springer-Verlag, Berlin, 1965.

# Chapter 7

## Center Manifold Analysis of the Delayed Liénard Equation

Siming Zhao and Tamás Kalmár-Nagy

**Abstract** In this chapter, the authors show the existence of the Hopf bifurcation in the delayed Liénard equation. The criterion for the criticality of the Hopf bifurcation is established based on the reduction of the infinite-dimensional problem onto a two-dimensional center manifold. Numerics based on DDE-Biftool are given to compare with the authors' theoretical calculation. The Liénard type sunflower equation is discussed as an illustrative example based on our method.

**Keywords:** Delayed Liénard Equation · Hopf bifurcation · Center manifold · Poincaré-Lyapunov constant

### 7.1 Introduction

Time delay is very common in many natural and engineering systems. It has been widely studied in fields as diverse as biology [1], population dynamics [2], neural networks [3], feedback controlled mechanical systems [4], machine tool vibrations [5,6], and lasers [7]. Delay effects can also be exploited to control nonlinear systems [8]. A good exposition of delay equations can be found in [9].

Several methods developed to study dynamical systems can be well applied to delay equations, these include the method of multiple scales [10], the Lindstedt-Poincaré method [11, 12], harmonic balance [13], and the averaging method [14].

Center manifold theory [15] is one of the rigorous mathematical tools to study bifurcations of delay-differential equations [16, 17]. In this chapter, we consider the Hopf bifurcation of the delayed Liénard equation

$$\ddot{x}(t) + f(x(t))\dot{x} + g(x(t - \tau)) = 0, \quad (7.1)$$

where  $f, g \in C^4$ ,  $f(0) = K > 0$ ,  $g(0) = 0$ ,  $g'(0) = 1$ , and  $\tau > 0$  is a finite time-delay.

Luk [18] and Zhang [19] derived necessary and sufficient conditions for the boundedness of all solutions and their derivatives of (7.1) using a Lyapunov-functional approach. Zhang [20] and Omari and Zangfin [21] showed the existence of periodic orbits for this equation. Metzen [22] proved the existence of periodic orbits in a duffing-type equation with delay and studied the effects of delay on the solvability of the problem. Acosta and Lizana [23] and Xu and Lu [24] investigated the linear stability of (7.1). Xu and Lu [24] performed a center manifold based Hopf bifurcation analysis. Campbell et al. [25] also used center manifold analysis of single and double Hopf bifurcations for a similar second-order delay differential equation. In these analysis, however, the curvature of the center manifold caused by quadratic terms was not accounted for. Colonius [26] studied optimal periodic control of (7.1).

As shown in [16], (7.1) can be rewritten in state space form as:

$$\begin{aligned}\dot{x}(t) &= y(t) - S(x(t)), \\ \dot{y}(t) &= -g(x(t - \tau)),\end{aligned}\tag{7.2}$$

where  $S(x) = \int_0^x f(\delta) d\delta$ . Expanding (7.2) in the neighborhood of the null solution up to third order yields:

$$\begin{aligned}\dot{x}(t) &= y(t) - Kx(t) + ax^2(t) + bx^3(t), \\ \dot{y}(t) &= -x(t - \tau) + cx^2(t - \tau) + dx^3(t - \tau),\end{aligned}\tag{7.3}$$

with  $a = -\frac{1}{2}f'(0)$ ,  $b = -\frac{1}{6}f''(0)$ ,  $c = -\frac{1}{2}g''(0)$ , and  $d = -\frac{1}{6}g'''(0)$ . By defining a new vector  $\mathbf{z} = (x, y)^T$ , the above system can be written as:

$$\dot{\mathbf{z}}(t) = \mathbf{Lz}(t) + \mathbf{Rz}(t - \tau) + \mathbf{f}(\mathbf{z}),\tag{7.4}$$

where

$$\mathbf{L} = \begin{pmatrix} -K & 1 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}, \quad \mathbf{f}(\mathbf{z}) = \begin{pmatrix} ax^2 + bx^3 \\ cx^2(t - T) + dx^3(t - T) \end{pmatrix}.\tag{7.5}$$

Note that there are many ways to write (7.1) in state space form. The reason why the form (7.2) is used is that when calculating the center manifold, the nonlinearity will only contain the first component of the state vector; this will simplify the calculation.

## 7.2 Linear Stability Analysis

The linear stability analysis of the  $\mathbf{z} = \mathbf{0}$  solution of (7.4) is based on the linear equation [24]

$$\dot{\mathbf{z}}(t) = \mathbf{Lz}(t) + \mathbf{Rz}(t - \tau).\tag{7.6}$$

The characteristic equation of (7.6) can be obtained by substituting the trial solution  $\mathbf{z}(t) = \mathbf{c} \exp^{\lambda t}$  into (7.6)

$$\lambda^2 + K\lambda + e^{-\lambda\tau} = 0. \quad (7.7)$$

On the stability boundary the characteristic equation has a pair of pure imaginary roots. To find these roots, we substitute  $\lambda = i\omega$ ,  $\omega > 0$  into (7.7) and separate the real and imaginary part to obtain

$$\begin{aligned} \omega^2 &= \cos \omega\tau, \\ K\omega &= \sin \omega\tau. \end{aligned} \quad (7.8)$$

Using simple trigonometry, (7.8) can be reduced to

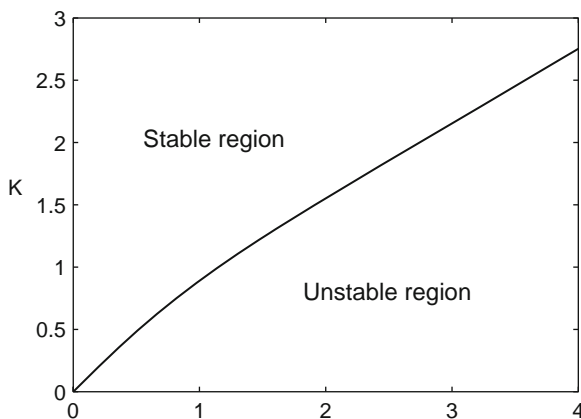
$$\begin{aligned} K &= \sqrt{\frac{1 - \omega^4}{\omega^2}}, \\ \tau &= \frac{1}{\omega} \left( 2n\pi + \arctan \frac{K}{\omega} \right), \end{aligned} \quad (7.9)$$

where  $n$  is a nonnegative integer. Since  $K > 0$ , we have  $0 < \omega < 1$ .

Note that (7.9) describes infinitely many curves in the  $(\tau, K)$  plane, but the first branch ( $n = 0$ ) (shown in Fig. 7.1) actually separates the stable and unstable regions (this is a consequence of a theorem by Hale [16]).

For a fixed time-delay  $\tau$ , the critical value of the bifurcation parameter  $K$  (i.e., on the stability boundary) will be denoted by  $k$ . A necessary condition for the existence of periodic orbits is that by changing  $K$  through  $k$ , the critical characteristic roots cross the imaginary axis with nonzero velocity, i.e.,  $d\mu/dK|_{K=k} \neq 0$ . Taking the first derivative with respect to  $K$  in (7.7) and using (7.9) gives

$$\gamma = \operatorname{Re} \frac{d\lambda}{dK} \Big|_{K=k} = \frac{\omega^2(2 + k\tau)}{(k - \tau\omega^2)^2 + (2\omega + k\tau\omega)^2} > 0. \quad (7.10)$$



**Fig. 7.1** Linear stability boundary of the delayed Liénard equation.

Therefore, the characteristic roots cross the imaginary axis with positive velocity. This velocity will later be used to calculate the estimation of the vibration amplitude.

In the following sections, we will use center manifold theory to investigate the Hopf bifurcation that might occur on the stability boundary and calculate the Poincaré-Lyapunov constant to characterize the criticality of the bifurcation. The solution method closely follows Kalmár-Nagy et al. [6].

### 7.3 Operator Differential Equation Formulation

Generally delay differential equations can be expressed as abstract evolution equations on the Banach space  $\mathcal{H}$  of continuously differentiable functions  $\mu : [-\tau, 0] \rightarrow \mathbb{R}^2$

$$\dot{\mathbf{z}}_t = \mathcal{D}\mathbf{z}_t + \mathcal{F}(\mathbf{z}_t), \quad (7.11)$$

where the shift of time  $\mathbf{z}_t(\varphi) \in \mathcal{H}$  is defined as:

$$\mathbf{z}_t(\varphi) = \mathbf{z}(t + \varphi), \quad \varphi \in [-\tau, 0]. \quad (7.12)$$

The linear operator  $\mathcal{D}$  at the critical bifurcation parameter assumes the form

$$\mathcal{D}\mathbf{u}(\varphi) = \begin{cases} \frac{d}{d\varphi}\mathbf{u}(\varphi), & \varphi \in [-\tau, 0) \\ \mathbf{L}\mathbf{u}(0) + \mathbf{R}\mathbf{u}(-\tau), & \varphi = 0 \end{cases},$$

while the nonlinear operator is written as

$$\begin{aligned} \mathcal{F}(\mathbf{u})(\varphi) &= \begin{cases} \mathbf{0}, & \varphi \in [-\tau, 0), \\ \mathbf{f}(\mathbf{u}), & \varphi = 0, \end{cases} \\ \mathbf{f}(\mathbf{u}(\varphi)) &= \begin{pmatrix} au_1^2(0) + bu_1^3(0) \\ cu_1^2(-\tau) + du_1^3(-\tau) \end{pmatrix}. \end{aligned}$$

To calculate the center manifold, we also need to define the adjoint space  $\mathcal{H}^*$  of continuously differentiable functions  $\theta : [0, \tau] \rightarrow \mathbb{R}^2$  together with the adjoint operator

$$\mathcal{D}^*\mathbf{u}(\theta) = \begin{cases} -\frac{d}{d\varphi}\mathbf{u}(\varphi), & \varphi \in (0, \tau] \\ \mathbf{L}^*\mathbf{u}(0) + \mathbf{R}^*\mathbf{u}(\tau), & \varphi = 0 \end{cases},$$

with respect to the bilinear form  $(\cdot, \cdot) : \mathcal{H}^* \times \mathcal{H} \rightarrow \mathbb{R}$

$$(v, \mu) = v^*(0)\mu(0) + \int_{-\tau}^0 v^*(\delta + \tau)\mathbf{R}\mu(\delta) d\delta. \quad (7.13)$$

These operators are used to construct a projection from the infinite-dimensional space onto the two-dimensional center manifold by using the eigenfunctions of the linear operator  $\mathcal{D}$  and its adjoint  $\mathcal{D}^*$ . The center subspace is spanned by the real and imaginary parts of the complex eigenfunction  $\mathbf{p}(\varphi)$  of  $\mathcal{D}$  corresponding to the

critical characteristic roots  $\pm i\omega$ . The complex eigenfunction  $\mathbf{p}(\varphi)$  and the eigenfunctions  $\mathbf{q}(\theta)$  of the adjoint operator  $\mathcal{D}^*$  can be found from:

$$\begin{aligned}\mathcal{D}\mathbf{p}(\varphi) &= i\omega\mathbf{p}(\varphi), \\ \mathcal{D}^*\mathbf{q}(\theta) &= -i\omega\mathbf{q}(\theta).\end{aligned}\tag{7.14}$$

Since the critical eigenvalues of the linear operator  $\mathcal{D}$  just coincide with the critical characteristic roots of the characteristic function  $D(\lambda, K)$ , the Hopf bifurcation can be studied at the two-dimensional center manifold embedded in the infinite dimensional phase space.

The general solutions to (7.14) are of the form:

$$\begin{aligned}\mathbf{p}(\varphi) &= \mathbf{c}e^{i\omega\varphi}, \\ \mathbf{q}(\theta) &= \mathbf{d}e^{i\omega\theta},\end{aligned}\tag{7.15}$$

and the constants  $\mathbf{p}, \mathbf{q}$  are found by using the boundary conditions embedded in the operator equations (7.14)

$$\begin{aligned}(i\omega\mathbf{I} - \mathbf{L} - e^{-i\omega\tau}\mathbf{R})\mathbf{c} &= \mathbf{0}, \\ (-i\omega\mathbf{I} - \mathbf{L}^T - e^{i\omega\tau}\mathbf{R}^T)\mathbf{d} &= \mathbf{0}.\end{aligned}$$

The vectors  $\mathbf{p}, \mathbf{q}$  should not be aligned, i.e., the bilinear form of  $\mathbf{q}$  and  $\mathbf{p}$  should be nonzero

$$(\mathbf{q}, \mathbf{p}) = \beta \neq 0.\tag{7.16}$$

The constant  $\beta$  can be freely chosen. Using (7.13) we have:

$$\begin{aligned}(\mathbf{q}, \mathbf{p}) &= \mathbf{q}^*(0)\mathbf{p}(0) + \int_{-\tau}^0 \mathbf{q}^*(\xi + \tau)\mathbf{R}\mathbf{p}(\xi) d\xi \\ &= \mathbf{d}^*\mathbf{c} + \mathbf{d}^*\mathbf{R}\mathbf{c}e^{-i\omega\tau} \int_{-\tau}^0 d\xi \\ &= \mathbf{d}^*(\mathbf{I} + \tau e^{-i\omega\tau}\mathbf{R})\mathbf{c}.\end{aligned}\tag{7.17}$$

To summarize, the vectors  $\mathbf{c}, \mathbf{d}$  are found from the following equations

$$(i\omega\mathbf{I} - \mathbf{L} - e^{-i\omega\tau}\mathbf{R})\mathbf{c} = \mathbf{0},\tag{7.18}$$

$$(i\omega\mathbf{I} + \mathbf{L}^T + e^{i\omega\tau}\mathbf{R}^T)\mathbf{d} = \mathbf{0},\tag{7.19}$$

$$\mathbf{d}^*(\mathbf{I} + \tau e^{-i\omega\tau}\mathbf{R})\mathbf{c} = 2.\tag{7.20}$$

There are three independent complex equations for four complex unknowns. (7.18) and (7.19) result in (here  $c_1, d_2$  are complex!)

$$\mathbf{c} = \begin{pmatrix} 1 \\ k + i\omega \end{pmatrix} c_1,\tag{7.21}$$

$$\mathbf{d} = \begin{pmatrix} -i\omega \\ 1 \end{pmatrix} d_2. \quad (7.22)$$

Then from (7.20) we obtain:

$$c_1 d_2^* [k - \omega^2 \tau + i(2\omega + k\omega\tau)] = 2. \quad (7.23)$$

We have the freedom to fix 1 unknown, and so we choose  $c_1 = 1$ , which in turn yields

$$d_2^* = \frac{2}{k - \omega^2 \tau + i(2\omega + k\omega\tau)}. \quad (7.24)$$

After separating the real and imaginary part

$$\mathbf{c}_1 = \text{Re } \mathbf{c} = \begin{pmatrix} c_{11} \\ c_{12} \end{pmatrix} = \begin{pmatrix} 1 \\ k \end{pmatrix},$$

$$\mathbf{c}_2 = \text{Im } \mathbf{c} = \begin{pmatrix} c_{21} \\ c_{22} \end{pmatrix} = \begin{pmatrix} 0 \\ \omega \end{pmatrix},$$

$$\mathbf{d}_1 = \text{Re } \mathbf{d} = \begin{pmatrix} d_{11} \\ d_{12} \end{pmatrix} = \Omega \begin{pmatrix} \omega^2(1 + \frac{1}{2}k\tau) \\ \frac{1}{2}(k - \omega^2\tau) \end{pmatrix},$$

$$\mathbf{d}_2 = \text{Im } \mathbf{d} = \begin{pmatrix} d_{21} \\ d_{22} \end{pmatrix} = \Omega \begin{pmatrix} -\frac{\omega}{2}(k - \omega^2\tau) \\ \omega(1 + \frac{1}{2}k\tau) \end{pmatrix},$$

with  $\Omega = \frac{4\gamma}{\omega^2(2+k\tau)}$ .

Let us decompose the solution  $\mathbf{z}_t(\varphi)$  into two components  $y_1(t)$  and  $y_2(t)$  lying in the center subspace and the infinite-dimensional component  $\mathbf{w}$  transverse to the center subspace:

$$\mathbf{z}_t(\varphi) = y_1(t)\mathbf{p}_1(\varphi) + y_2(t)\mathbf{p}_2(\varphi) + \mathbf{w}(t)(\varphi), \quad (7.25)$$

$$y_1(t) = (\mathbf{q}_1, \mathbf{z}_t) |_{\varphi=0}, \quad y_2(t) = (\mathbf{q}_2, \mathbf{z}_t) |_{\varphi=0}.$$

With these new coordinates the operator differential equation (7.11) is be transformed into:

$$\dot{y}_1 = \omega y_2 + \mathbf{q}_1^T(0)\mathbf{F}, \quad (7.26)$$

$$\dot{y}_2 = -\omega y_1 + \mathbf{q}_2^T(0)\mathbf{F}, \quad (7.27)$$

$$\dot{\mathbf{w}} = \mathcal{D}\mathbf{w} + \mathcal{F}(\mathbf{z}_t) - \mathbf{q}_1^T(0)\mathbf{F}\mathbf{p}_1 - \mathbf{q}_2^T(0)\mathbf{F}\mathbf{p}_2. \quad (7.28)$$

Note that the nonlinear operator in (7.28) should be written as:

$$\mathcal{F}(y_1\mathbf{p}_1 + y_2\mathbf{p}_2 + \mathbf{w}) = \begin{cases} \mathbf{0}, & \varphi \in [-T, 0) \\ \mathbf{F}, & \varphi = 0 \end{cases},$$

where  $\mathbf{F} = (f_1, f_2)^T$  and  $f_1$  and  $f_2$  are given as (neglecting terms higher than third order):

$$f_1 = a(w_1(0) + y_1)^2 + b(w_1(0) + y_1)^3 = a(y_1^2 + 2y_1w_1(0)) + by_1^3, \quad (7.29)$$

$$\begin{aligned}
f_2 &= c(w_1(-\tau) + \cos \omega \tau y_1 - \sin \omega \tau y_2)^2 + d(w_1(-\tau) + \cos \omega \tau y_1 - \sin \omega \tau y_2)^3 \\
&= c(\omega^4 y_1^2 + k^2 \omega^2 y_2^2 - 2k\omega^3 y_1 y_2 + 2\omega^2 w_1(-\tau) y_1 - 2k\omega w_1(-\tau) y_2) \\
&\quad + d(\omega^6 y_1^3 - k^3 \omega^3 y_2^3 - 3k\omega^5 y_1^2 y_2 + 3k^2 \omega^4 y_1 y_2^2).
\end{aligned} \tag{7.30}$$

In the next section we will approximate  $\mathbf{w}(y_1, y_2)(\varphi)$  assuming that it only contains quadratic terms (higher order terms of  $\mathbf{w}$  are not relevant for local Hopf bifurcation analysis).

## 7.4 Center Manifold Reduction

The center manifold is tangent to  $y_1, y_2$  plane at the origin, and it is locally invariant and attractive to the flow of the system (7.11). Notice that when  $a = c = 0$  (symmetric nonlinearities), the center manifold coincides with the center subspace, which is spanned by the eigenfunctions calculated earlier. Since the nonlinearities considered here are not always symmetric, the second-order expansion of the center manifold needs to be computed. Neglecting higher order terms for the local bifurcation analysis

$$\mathbf{w}(y_1, y_2)(\varphi) = \frac{1}{2} \left( \mathbf{h}_1(\varphi) y_1^2 + 2\mathbf{h}_2(\varphi) y_1 y_2 + \mathbf{h}_3(\varphi) y_2^2 \right). \tag{7.31}$$

The time derivative of  $\varphi$  can be expressed by differentiating the right-hand side of the above equation via substituting (7.26) and (7.27):

$$\begin{aligned}
\dot{\mathbf{w}} &= \mathbf{h}_1 y_1 \dot{y}_1 + \mathbf{h}_2 y_2 \dot{y}_1 + \mathbf{h}_2 y_1 \dot{y}_2 + \mathbf{h}_3 y_2 \dot{y}_2 \\
&= \dot{y}_1 (\mathbf{h}_1 y_1 + \mathbf{h}_2 y_2) + \dot{y}_2 (\mathbf{h}_2 y_1 + \mathbf{h}_3 y_2) \\
&= -\omega \mathbf{h}_2 y_1^2 + \omega (\mathbf{h}_1 - \mathbf{h}_3) y_1 y_2 + \omega \mathbf{h}_2 y_2^2 + O(y^3).
\end{aligned} \tag{7.32}$$

Comparing the coefficients of  $y_1^2, y_1 y_2, y_2^2$  with another form of  $\dot{\mathbf{w}}$  from (7.28) we arrive at the following boundary value problem:

$$\begin{aligned}
\frac{1}{2} \dot{\mathbf{h}}_1 &= -\omega \mathbf{h}_2 + \mathbf{m} f_{111} + \mathbf{n} f_{211}, \\
\dot{\mathbf{h}}_2 &= \omega \mathbf{h}_1 - \omega \mathbf{h}_3 + \mathbf{m} f_{112} + \mathbf{n} f_{212}, \\
\frac{1}{2} \dot{\mathbf{h}}_3 &= \omega \mathbf{h}_2 + \mathbf{m} f_{122} + \mathbf{n} f_{222},
\end{aligned} \tag{7.33}$$

$$\begin{aligned}
\frac{1}{2} \left( \mathbf{L} \mathbf{h}_1(0) + \mathbf{R} \mathbf{h}_1(-\tau) \right) &= -\omega \mathbf{h}_2(0) + \mathbf{m}(0) f_{111} + \mathbf{n}(0) f_{211} - \mathbf{s}_1, \\
\mathbf{L} \mathbf{h}_2(0) + \mathbf{R} \mathbf{h}_2(-\tau) &= \omega \mathbf{h}_1(0) - \omega \mathbf{h}_3(0) + \mathbf{m}(0) f_{112} + \mathbf{n}(0) f_{212} - \mathbf{s}_2, \\
\frac{1}{2} \left( \mathbf{L} \mathbf{h}_3(0) + \mathbf{R} \mathbf{h}_3(-\tau) \right) &= \omega \mathbf{h}_2(0) + \mathbf{m}(0) f_{122} + \mathbf{n}(0) f_{222} - \mathbf{s}_3,
\end{aligned} \tag{7.34}$$

with

$$\begin{aligned}\mathbf{m}(\varphi) &= d_{11}\mathbf{p}_1(\varphi) + d_{21}\mathbf{p}_2(\varphi), \\ \mathbf{n}(\varphi) &= d_{12}\mathbf{p}_1(\varphi) + d_{22}\mathbf{p}_2(\varphi).\end{aligned}\tag{7.35}$$

The coefficients of the quadratic terms can be extracted from (7.29) and (7.30) as:

$$\begin{aligned}f_{111} &= a, & f_{211} &= c\omega^4, \\ f_{112} &= 0, & f_{212} &= -2ck\omega^3, \\ f_{122} &= 0, & f_{222} &= ck^2\omega^2.\end{aligned}$$

By introducing the following notation

$$\begin{aligned}\mathbf{h} &= \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{pmatrix}, & \mathbf{C}_{6 \times 6} &= \omega \begin{pmatrix} \mathbf{0} & -2\mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & 2\mathbf{I} & \mathbf{0} \end{pmatrix}, \\ \mathbf{s} &= \begin{pmatrix} 2\mathbf{S}_0\mathbf{s}_1 \\ \mathbf{S}_0\mathbf{s}_2 \\ 2\mathbf{S}_0\mathbf{s}_3 \end{pmatrix}, & \mathbf{n} &= \begin{pmatrix} 2\mathbf{N}_0\mathbf{s}_1 \\ \mathbf{N}_0\mathbf{s}_2 \\ 2\mathbf{N}_0\mathbf{s}_3 \end{pmatrix}, \\ \mathbf{s}_1 &= \begin{pmatrix} f_{111} \\ f_{211} \end{pmatrix}, & \mathbf{s}_2 &= \begin{pmatrix} f_{112} \\ f_{212} \end{pmatrix}, & \mathbf{s}_3 &= \begin{pmatrix} f_{122} \\ f_{222} \end{pmatrix}, \\ \mathbf{S}_0 &= \begin{pmatrix} d_{11} & d_{12} \\ kd_{11} + \omega d_{21} & kd_{12} + \omega d_{22} \end{pmatrix}, \\ \mathbf{N}_0 &= \begin{pmatrix} d_{21} & d_{22} \\ kd_{21} - \omega d_{11} & kd_{22} - \omega d_{12} \end{pmatrix}.\end{aligned}$$

Eq. (7.33) is written as the inhomogeneous differential equation:

$$\frac{d}{d\varphi}\mathbf{h}(\varphi) = \mathbf{C}\mathbf{h} + \mathbf{s} \cos \omega\varphi + \mathbf{n} \sin \omega\varphi.\tag{7.36}$$

This ODE has the general solution form:

$$\mathbf{h}(\varphi) = e^{\mathbf{C}\varphi}\mathbf{K} + \mathbf{M} \cos \omega\varphi + \mathbf{N} \sin \omega\varphi.\tag{7.37}$$

After substituting this solution form into (7.36) we get the following equations for solving matrix  $\mathbf{M}$  and  $\mathbf{N}$  together with the corresponding boundary value problem for solving  $\mathbf{K}$

$$\begin{pmatrix} \mathbf{C}_{6 \times 6} & -\omega\mathbf{I}_{6 \times 6} \\ \omega\mathbf{I}_{6 \times 6} & \mathbf{C}_{6 \times 6} \end{pmatrix} \begin{pmatrix} \mathbf{M} \\ \mathbf{N} \end{pmatrix} = - \begin{pmatrix} \mathbf{s} \\ \mathbf{n} \end{pmatrix},\tag{7.38}$$

$$\mathbf{P}\mathbf{h}(0) + \mathbf{Q}\mathbf{h}(-\tau) = \mathbf{s} - \mathbf{r},\tag{7.39}$$

where

$$\mathbf{P} = \begin{pmatrix} \mathbf{L} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{L} \end{pmatrix} - \mathbf{C}_{6 \times 6},$$

$$\mathbf{Q} = \begin{pmatrix} \mathbf{R} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R} \end{pmatrix}, \quad \mathbf{r} = \begin{pmatrix} 2\mathbf{s}_1 \\ \mathbf{s}_2 \\ 2\mathbf{s}_3 \end{pmatrix}.$$

The expressions for  $\mathbf{w}_1(0)$  and  $\mathbf{w}_1(-\tau)$  are given as:

$$\begin{aligned} \mathbf{w}_1(0) &= \frac{1}{2} \left( (M_1 + K_1)y_1^2 + 2(M_3 + K_3)y_1y_2 + (M_5 + K_5)y_2^2 \right) \\ &= h_{110}y_1^2 + h_{210}y_1y_2 + h_{310}y_2^2, \end{aligned} \quad (7.40)$$

$$\begin{aligned} \mathbf{w}_1(-\tau) &= \frac{1}{2} \left( (e^{-\mathbf{C}\tau}\mathbf{K}|_1 + M_1 \cos \omega\tau - N_1 \sin \omega\tau)y_1^2 \right. \\ &\quad + 2(e^{-\mathbf{C}\tau}\mathbf{K}|_3 + M_3 \cos \omega\tau - N_3 \sin \omega\tau)y_1y_2 \\ &\quad \left. + (e^{-\mathbf{C}\tau}\mathbf{K}|_5 + M_5 \cos \omega\tau - N_5 \sin \omega\tau)y_2^2 \right) \\ &= h_{11\tau}y_1^2 + h_{21\tau}y_1y_2 + h_{31\tau}y_2^2. \end{aligned} \quad (7.41)$$

From (7.38) and (7.39) we can calculate the first, third, and fifth component of  $\mathbf{M}$ ,  $\mathbf{N}$ ,  $\mathbf{K}$ ,  $e^{-\mathbf{C}\tau}\mathbf{K}$

$$\begin{aligned} \begin{pmatrix} M_1 \\ M_3 \\ M_5 \end{pmatrix} &= -\frac{2}{3\omega} \begin{pmatrix} ad_{21} + c\omega^2(-2d_{12}k\omega + d_{22}(2k^2 + \omega^2)) \\ -ad_{11} + c\omega^2(d_{22}k\omega + d_{12}(k^2 - \omega^2)) \\ 2ad_{21} + c\omega^2(2d_{12}k\omega + d_{22}(k^2 + 2\omega^2)) \end{pmatrix}, \\ \begin{pmatrix} N_1 \\ N_3 \\ N_5 \end{pmatrix} &= \frac{2}{3\omega} \begin{pmatrix} ad_{11} + c\omega^2(2d_{22}k\omega + d_{12}(2k^2 + \omega^2)) \\ ad_{21} + c\omega^2(d_{12}k\omega + d_{22}(-k^2 + \omega^2)) \\ 2ad_{11} + c\omega^2(-2d_{22}k\omega + d_{12}(k^2 + 2\omega^2)) \end{pmatrix}, \\ \begin{pmatrix} K_1 \\ K_3 \\ K_5 \end{pmatrix} &= \zeta \begin{pmatrix} \omega(-2ak(\omega^2 - 1) + 3c\omega^2(4 + k^4 - 2\omega^2 - \omega^4)) \\ a(1 - 4\omega^2 - 2k^2\omega^2) + ck(1 + 2\omega^4) \\ \omega(2ak(-1 + \omega^2) + c(2k^2 + 8\omega^2 - 2\omega^4)) \end{pmatrix}, \\ &\quad \begin{pmatrix} e^{-\mathbf{C}\tau}\mathbf{K}|_1 \\ e^{-\mathbf{C}\tau}\mathbf{K}|_3 \\ e^{-\mathbf{C}\tau}\mathbf{K}|_5 \end{pmatrix} = \\ &\quad \zeta \begin{pmatrix} \omega(-2ak(1 + 2\omega^4) + c\omega^2(9 + 2k^2 + 2k^4 + (2k^2 - 6)\omega^2 + 8k^2\omega^4)) \\ a(1 - 4\omega^6) + ck\omega^2(-1 - 5k^2 + (7 + 4k^4)\omega^2 - 4k^2\omega^4) \\ \omega(2ak(1 + 2\omega^4) + c(3k^2 - 2 + (8 - 8k^2)\omega^2 + 8k^4\omega^4)) \end{pmatrix}, \end{aligned}$$

where  $\zeta = \frac{2\omega}{5 + 12\omega^4 - 8\omega^6}$ .

## 7.5 Hopf Bifurcation Analysis

To restrict a third-order approximation of system (7.26) and (7.27) to the two-dimensional center manifold calculated in the previous section, the dynamics of  $y_1$  and  $y_2$  is assumed to have the form:

$$\begin{aligned}\dot{y}_1 &= \omega y_2 + a_{20}y_1^2 + a_{11}y_1y_2 + a_{02}y_2^2 + a_{30}y_1^3 + a_{21}y_1^2y_2 + a_{12}y_1y_2^2 + a_{03}y_2^3, \\ \dot{y}_2 &= -\omega y_1 + b_{20}y_1^2 + b_{11}y_1y_2 + b_{02}y_2^2 + b_{30}y_1^3 + b_{21}y_1^2y_2 + b_{12}y_1y_2^2 + b_{03}y_2^3.\end{aligned}\quad (7.42)$$

Using the 10 out of these 14 coefficients  $a_{jk}, b_{jk}$ , the so called Poincaré-Lyapunov constant  $\Delta$  can be calculated as shown in [15]

$$\begin{aligned}\Delta &= \frac{1}{8\omega}((a_{20} + a_{02})(-a_{11} + b_{20} - b_{02}) + (b_{20} + b_{02})(a_{20} - a_{02} + b_{11})) \\ &\quad + \frac{1}{8}(3a_{30} + a_{12} + b_{21} + 3b_{03}).\end{aligned}\quad (7.43)$$

Based on the center manifold calculation, the ten coefficients are as follows:

$$\begin{aligned}a_{20} &= ad_{11} + c\omega^4 d_{12}, \\ b_{20} &= ad_{21} + c\omega^4 d_{22}, \\ a_{11} &= -2ck\omega^3 d_{12}, \\ b_{11} &= -2ck\omega^3 d_{22}, \\ a_{02} &= ck^2\omega^2 d_{12}, \\ b_{02} &= ck^2\omega^2 d_{22}, \\ a_{30} &= (2ah_{110} + b)d_{11} + (2c\omega^2 h_{11\tau} + d\omega^6)d_{12}, \\ a_{12} &= 2ah_{310}d_{11} + (2c\omega^2 h_{31\tau} - 2ck\omega h_{21\tau} + 3dk^2\omega^4)d_{12}, \\ b_{21} &= 2ah_{210}d_{21} + (2c\omega^2 h_{21\tau} - 2ck\omega h_{11\tau} - 3dk\omega^5)d_{22}, \\ b_{03} &= -(2ck\omega h_{31\tau} + dk^3\omega^3)d_{22}.\end{aligned}$$

Substitute all these coefficients into (7.43) and tedious simplification can lead us to:

$$\Delta = l_1 d_{12} + l_2 d_{22}, \quad (7.44)$$

where we have

$$\begin{aligned}l_1 &= \frac{3}{8}d\omega^2 + \frac{a^2}{4}\omega\zeta(1 + 4\omega^2 - 2\omega^4) - \frac{ac}{2}k\omega\zeta(1 + \omega^2 + \omega^4) \\ &\quad + \frac{c^2}{4}\omega\zeta\left(\frac{11}{2} + k^2 + 2\omega^2 + 12\omega^4 - 12\omega^6\right), \\ l_2 &= \frac{3}{8}b\omega - \frac{3}{8}dk\omega + \frac{a^2}{2}k\omega^2\zeta(1 - \omega^2) + \frac{ac}{4}\zeta\left(\frac{7}{2} + \omega^2 + 10\omega^4 - 10\omega^6\right) \\ &\quad + \frac{c^2}{4}k\zeta\left(-\frac{11}{2} + \omega^2 - 12\omega^4 + 12\omega^6\right).\end{aligned}\quad (7.45)$$

The negative or positive sign of  $\Delta$  determines whether the Hopf bifurcation is supercritical or subcritical.

## 7.6 Numerical Results

To validate the results of the center manifold calculations, we used the continuation-based DDE-Biftool [27, 28] as well as the MATLAB numerical solver DDE-23. By defining  $\alpha = \frac{\gamma}{\Delta}$ , the vibration amplitude in the neighborhood of the bifurcation point is estimated as:

$$r = \sqrt{\alpha(K - k)}. \quad (7.46)$$

The delayed Liénard equation

$$\begin{aligned} \dot{x}(t) &= y(t) - Kx(t) + ax^2(t) + bx^3(t), \\ \dot{y}(t) &= -x(t - \tau) + cx^2(t - \tau) + dx^3(t - \tau), \end{aligned} \quad (7.47)$$

with three different sets of parameters (see Table 7.1) was solved by continuation (DDE-Biftool) and numerical integration (DDE-23). For  $\tau = 1, 2, 3, 4$  Table 7.2 shows the value of  $k$  at the bifurcation point, the critical frequency  $\omega$ , and the percent error  $\varepsilon = 100 \left| \frac{\alpha - \alpha_{\text{num}}}{\alpha} \right|$ . The numerical approximation  $\alpha_{\text{num}}$  has been obtained from the DDE-Biftool results (using amplitudes corresponding to values of the bifurcation parameter  $K$  such that  $|K - k| \leq 0.0005k$ ) by least-squares fit.

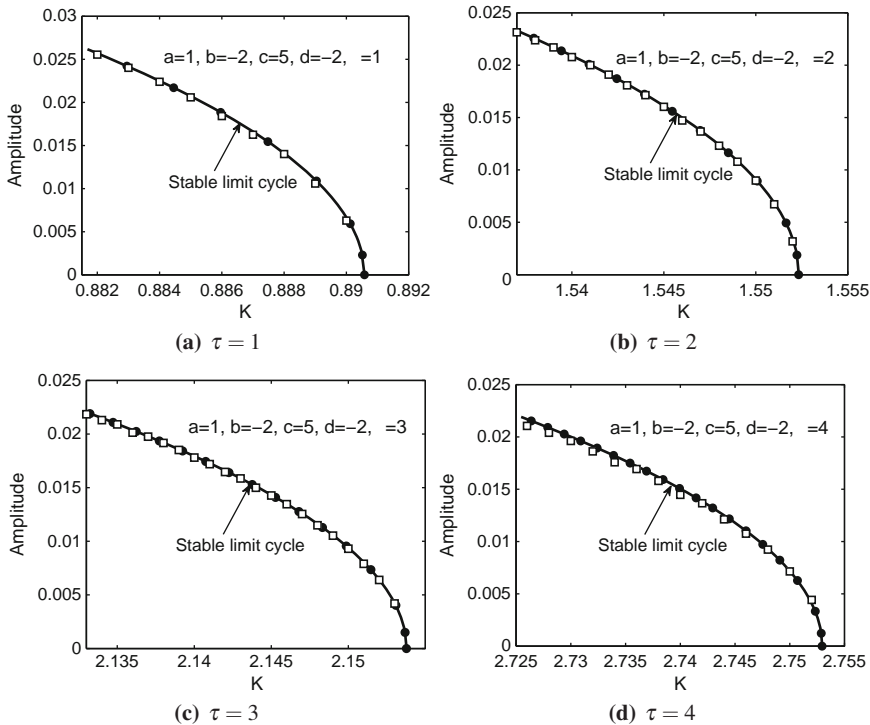
Figures 7.2–7.4 show the bifurcation diagrams of (7.47) for the above parameter values. The solid line, dots, and rectangles show the analytical amplitude estimate (7.46), the DDE-Biftool results, and numerical results by DDE-23, respectively. The DDE-23 results are obtained by combining numerical integration, estimation of amplitude decay/growth, and bisection to locate periodic orbits. Finally, by

**Table 7.1** Sets of parameter values of three different delayed Liénard equations

	a	b	c	d
I	1	-2	5	-2
II	-3	-2	1	-4
III	-3	-1	4	3

**Table 7.2** Value of the bifurcation parameter  $K$  on the stability boundary, critical frequency  $\omega$ , and error  $\varepsilon$  evaluated at different time delays  $\tau$

$\tau$	1	2	3	4
$k$	0.891	1.552	2.154	2.753
$\omega$	0.824	0.601	0.454	0.360
$\varepsilon_{\text{I}}$	0.13%	0.10%	0.15%	0.11%
$\varepsilon_{\text{II}}$	0.91%	0.94%	0.30%	0.25%
$\varepsilon_{\text{III}}$	0.29%	0.15%	0.09%	0.15%



**Fig. 7.2** Bifurcation diagram for  $K$  for case I of Table 7.2. The *solid line* is the estimated vibration amplitude based on (7.46), the *dots* and *rectangles* are results obtained from DDE-Biftool and DDE-23, respectively. The bifurcation is subcritical.

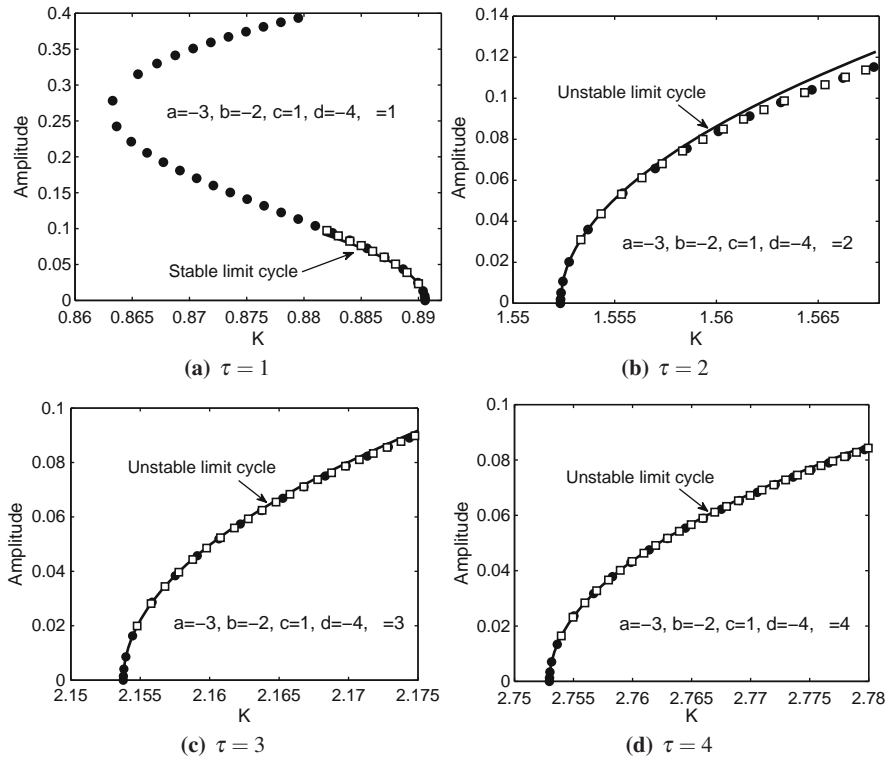
randomly choosing  $a, b, c, d$  from  $[-10, 10]$  and  $\tau$  from  $[0, 5]$ , 1,000 DDE-Biftool simulations were performed and the amplitude results were compared with the analytical ones. We found that  $\alpha$  agrees very well with  $\alpha_{\text{num}}$  (approximation is again based on amplitudes within 0.05% of the critical value  $k$ ): the mean error is 0.8%, with a small variance of  $5.43 \times 10^{-4}$ .

Having verified our analytical results, we can also utilize them to study the so-called Sunflower equation.

## 7.7 Hopf Bifurcation in the Sunflower Equation

Israelson and Johnson [29] proposed the equation:

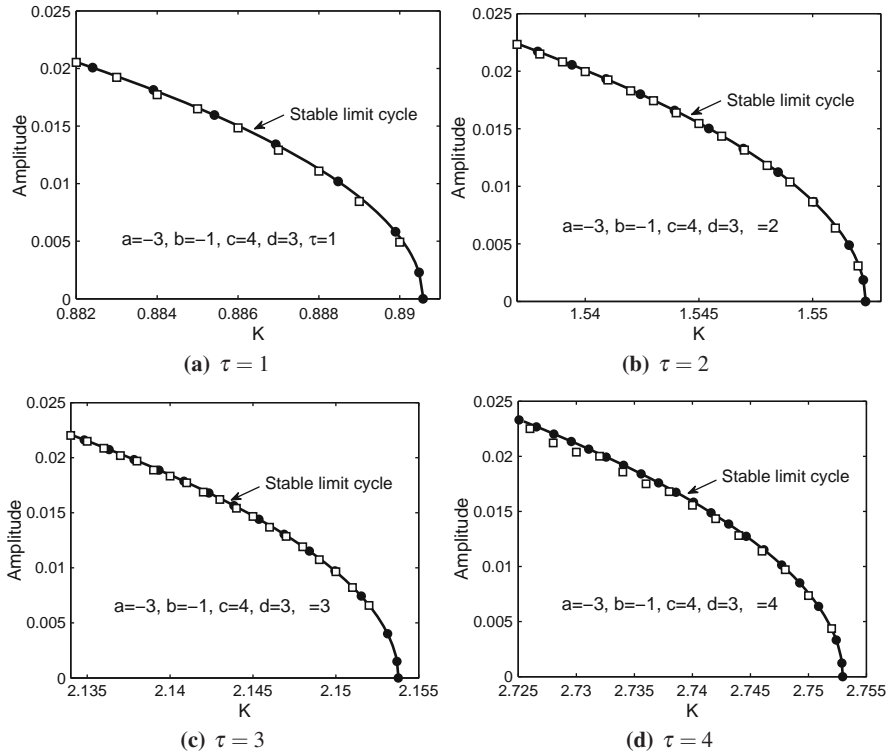
$$y'' + \frac{A}{\varepsilon} y' + \frac{B}{\varepsilon} \sin y(\tilde{t} - \varepsilon) = 0 \quad (7.48)$$



**Fig. 7.3** Bifurcation diagram for  $K$  for case II of Table 7.2. The *solid line* is the estimated vibration amplitude based on (7.46) the *dots* and *rectangles* are results obtained from DDE-Biftool and DDE-23, respectively. The bifurcation is subcritical.

to explain the helical movement of the tip of a growing plant. The upper part of the stem of the sunflower performs a rotating movement.  $y(\tilde{t})$  is the angle of the plant with respect to the vertical line, the delay factor  $\varepsilon$  is corresponding to a geotropic reaction time in the effect due to accumulation of the growth hormone alternatively on both side of the plant. The parameters  $A$  and  $B$  can be obtained experimentally.

Somolinos [30] proved the existence of periodic solutions for (7.48); this result covers both small amplitude limit cycle generated by Hopf bifurcation and large amplitude limit cycles. Casal and Freedman [11] computed a perturbation expansion based on the Lindstedt-Poincaré method. MacDonald [13] performed first-order harmonic balance of (7.48). Recently, Liu and Kalmár-Nagy [31] used the high-dimensional harmonic balance technique to compute limit cycle amplitudes and frequencies.



**Fig. 7.4** Bifurcation diagram for  $K$  for case III of Table 7.2. The *solid line* is the estimated vibration amplitude based on (7.46) the *dots* and *rectangles* are results obtained from DDE-Biftool and DDE-23, respectively. The bifurcation is supercritical.

Introducing a time scaling  $t \rightarrow \sqrt{\frac{B}{\varepsilon}} \tilde{t}$  and expanding (7.48) about the null solution up to third order yields:

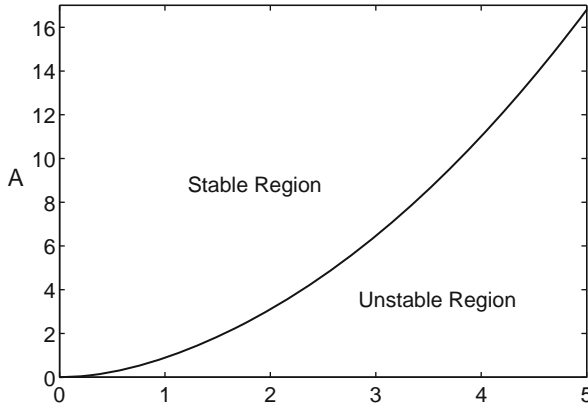
$$\ddot{x} + \frac{A}{\tau} \dot{x} + x(t - \tau) - \frac{1}{6} x^3(t - \tau) = 0, \quad (7.49)$$

where  $\tau = \sqrt{B\varepsilon}$ ,  $x(t) = y(\sqrt{\frac{\varepsilon}{B}} \tilde{t})$ . This equation is in the same form as (7.1) with  $K = \frac{A}{\tau}$ ,  $a = b = c = 0$ , and  $d = \frac{1}{6}$ ; therefore, our previous results can be directly applied. The characteristic equation is:

$$\lambda^2 + \frac{A}{\tau} \lambda + e^{-\lambda\tau} = 0. \quad (7.50)$$

On the stability boundary

$$\begin{aligned} \omega^2 &= \cos \omega \tau, \\ \frac{A_{\text{cr}}}{\tau} \omega &= \sin \omega \tau. \end{aligned} \quad (7.51)$$



**Fig. 7.5** Linear stability boundary of the Sunflower equation.

The parametric curve  $(\tau(\omega), A_{\text{cr}}(\omega))$  describes the stability boundary (see Fig. 7.5). We now substitute  $\frac{A_{\text{cr}}}{\tau} = k$  into our result for the Poincaré-Lyapunov formula (7.44) to yield

$$\Delta = -\frac{\Omega}{32\tau}(A_{\text{cr}}\omega^2 + \tau^2) < 0, \quad (7.52)$$

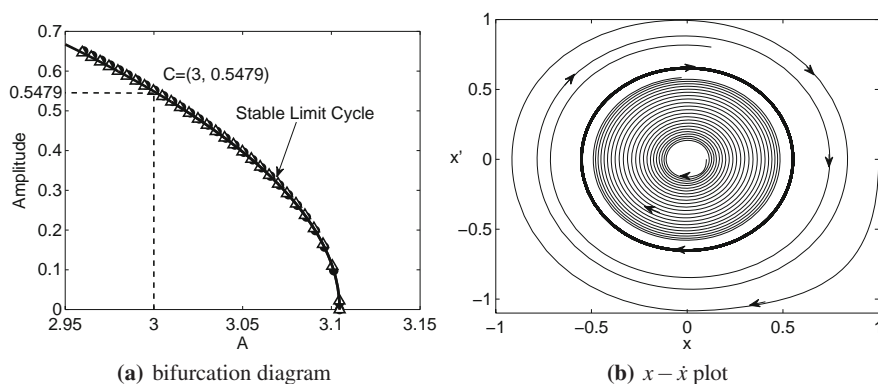
where  $\Omega = \frac{4\tau^2}{(A_{\text{cr}} - \tau^2\omega^2)^2 + (2\tau\omega + A_{\text{cr}}\tau\omega)^2}$ . To obtain the amplitude estimate, we also need to calculate root crossing velocity through bifurcating A:

$$\gamma = \text{Re} \frac{d\lambda}{dA} \big|_{A=A_{\text{cr}}, T=\tau} = -\frac{\omega^2}{4\tau}(2 + A_{\text{cr}})\Omega < 0. \quad (7.53)$$

We conclude that the Hopf bifurcation of the Sunflower equation is always supercritical. This conclusion is in full agreement with that from the earlier studies referred to. From (7.46), (7.52), and (7.53), the amplitude of the limit cycle can be estimated as:

$$r = 2\omega \sqrt{\frac{-2(2 + A_{\text{cr}})(A - A_{\text{cr}})}{A_{\text{cr}}\omega^2 + \tau^2}}. \quad (7.54)$$

Figure 7.6a shows the amplitude estimate for  $B = 4$  and  $\varepsilon = 1$ . The solid line denotes the plot of the analytical result (7.54), while the dots and triangles correspond to the numerical results from DDE-Biftool based on the original equation (7.48) and the Taylor expanded one (7.49). Figure 7.6b shows the  $x - \dot{x}$  plot corresponding to point C ( $A = 3, B = 4, \varepsilon = 1$ ) in Fig. 7.6a.



**Fig. 7.6** (a) shows the analytical amplitude estimate and DDE-Biftool simulation of (7.49) and (7.48) when  $B = 4, \varepsilon = 1$ . (b) shows the  $x - \dot{x}$  plot of the point C in (a). The initial function is chosen as  $x(t) = 0.1, \quad t \in (-1, 0]$ .

## 7.8 Concluding Remarks

In this chapter, we proved the existence of the Hopf bifurcation of the null solution of the delayed Liénard equation using center manifold analysis. Based on the reduction onto a two-dimensional system, we have presented a closed-form criterion for the criticality of the Hopf bifurcation. The amplitude estimate for the bifurcating limit cycle was obtained by using the calculated root crossing velocity ( $\gamma$ ) and Poincaré-Lyapunov constant ( $\Delta$ ). The analytical result agrees well with the numerical results obtained from DDE-Biftool and DDE-23. The Sunflower equation is investigated as a special case of the delayed Liénard equation.

## References

1. N. MacDonald. *Biological Delay Systems*. Cambridge University Press, New York, 1989.
2. Y. Kuang. *Delay Differential Equations with Applications in Population Dynamics*. *Mathematics in Science and Engineering*, 191, 1993.
3. A. Beuter, J. Bélair, C. Labrie, and J. Bélair. Feedback and Delays in Neurological Diseases: A Modeling Study Using Dynamical Systems. *Bulletin of Mathematical Biology*, 55(3): 525–541, 1993.
4. H.Y. Hu. *Dynamics of Controlled Mechanical Systems with Delayed Feedback*. Springer, New York, 2002.
5. G. Stépán. Modelling Nonlinear Regenerative Effects in Metal Cutting. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 359(1781):739–757, 2001.
6. T. Kalmár-Nagy, G. Stépán, and F.C. Moon. Subcritical Hopf Bifurcation in the Delay Equation Model for Machine Tool Vibrations. *Nonlinear Dynamics*, 26(2):121–142, 2001.
7. D. Pieroux, T. Erneux, T. Luzyanina, and K. Engelborghs. Interacting Pairs of Periodic Solutions Lead to Tori in Lasers Subject to Delayed Feedback. *Physical Review E*, 63(3):36211, 2001.

8. K. Pyragas. Continuous Control of Chaos by Self-controlling Feedback. *Physics Letters A*, 170(6):421–428, 1992.
9. G. Stépán. *Retarded Dynamical Systems: Stability and Characteristic Functions*. Longman Scientific & Technical, Marlow, New York, 1989.
10. H.Y. Hu, E.H. Dowell, and L.N. Virgin. Resonances of a Harmonically Forced Duffing Oscillator with Time Delay State Feedback. *Nonlinear Dynamics*, 15(4):311–327, 1998.
11. A. Casal and M.M. Freedman. A Poincaré-Lindstedt Approach to Bifurcation Problems for Differential-delay Equations. *IEEE Transactions on Automatic Control*, 25(5):967–973, 1980.
12. H.C. Morris. A Perturbative Approach to Periodic Solutions of Delay-differential Equations. *IMA Journal of Applied Mathematics*, 18(1):15–24, 2001.
13. N. MacDonald. Harmonic Balance in Delay-differential Equations. *Journal of Sound and Vibration*, 186(4):649–656, 1995.
14. B. Lehman and S. Weibel. Moving Averages for Periodic Delay Differential and Difference equations. *Lecture Notes in Control and Information Science*, 228:158–183.
15. B.D. Hassard, N.D. Kazarinoff, and Y.H. Wan. Theory and Applications of Hopf Bifurcation. *London Mathematical Society Lecture Note Series*, 41, 1981.
16. J.K. Hale. Theory of Functional Differential Equations. *Applied Mathematical Sciences*, vol. 3, 1977.
17. J.K. Hale. Nonlinear Oscillations in Equations with Time delay, in *Nonlinear Oscillations in Biology*, ed. Hoppenstadt, K. (AMS, Providence, RI) 157–185, 1979.
18. W. Luk. Some Results Concerning the Boundedness of Solutions of Liénard Equations With Delay. *SIAM Journal on Applied Mathematics*, 30(4):768–774, 1976.
19. B. Zhang. On the Retarded Liénard Equation. *Proceedings of the American Mathematical Society*, 115(3):779–785, 1992.
20. B. Zhang. Periodic Solutions of the Retarded Liénard Equation. *Annali di Matematica Pura ed Applicata*, 172(1):25–42, 1997.
21. P. Omari and F. Zanolin. Periodic Solutions of Liénard Equations. *Rendiconti del Seminario Matematico della Università di Padova*, 72:203–230, 1984.
22. G. Metzen. Existence of Periodic Solutions of Second Order Differential Equations with Delay. *Proceedings of the American Mathematical Society*, 103(3):765–772, 1988.
23. A. Acosta and M. Lizana. Hopf Bifurcation for the Equation  $x''(t) + f(x(t))x'(t) + g(x(t - r)) = 0$ . *Divulgaciones Matemáticas*, 13:35–43, 2005.
24. J. Xu and Q.S. Lu. Hopf Bifurcation of Time-delay Liénard Equations. *International Journal of Bifurcation and Chaos*, 9(5):939–951, 1999.
25. S.A. Campbell, J. Bélair, T. Ohira, and J. Milton. Limit cycles, Tori, and Complex Dynamics in a Second-order Differential Equation with Delayed Negative Feedback. *Journal of Dynamics and Differential Equations*, 7(1):213–236, 1995.
26. F. Colonius and F. Box. Optimal Periodic Control of Retarded Liénard Equation. *Distributed Parameter Systems, Proceedings of Science International Conference, Vorau/Austria*, 1984.
27. K. Engelborghs, T. Luzyanina, and D. Roose. Numerical Bifurcation Analysis of Delay Differential Equations Using DDE-BIFTOOL. *ACM Transactions on Mathematical Software*, 28(1):1–21, 2002.
28. K. Engelborghs, T. Luzyanina, and G. Samaey. DDE-BIFTOOL v. 2.00: A Matlab Package for Bifurcation Analysis of Delay Differential Equations. *Report TW*, 330, 2001.
29. D. Israelson and A. Johnson. Theory of Circumnutations in *Helianthus Annus*. *Physiology Plant*, 20:957–976, 1967.
30. A.S. Somolinos. Periodic Solutions of the Sunflower Equation:  $x + (a/r)x + (b/r)\sin x(t - r) = 0$ . *Q. Appl. Math. (Quarterly of Applied Mathematics)* 35:465–478, 1978.
31. L.P. Liu and T. Kalmár-Nagy. High Dimensional Harmonic Balance Analysis for Second-Order Delay-Differential Equations. *Journal of Vibration and Control*, in press.

# Chapter 8

## Calculating Center Manifolds for Delay Differential Equations Using Maple™

Sue Ann Campbell

**Abstract** In this chapter, we demonstrate how a symbolic algebra package may be used to compute the center manifold of a nonhyperbolic equilibrium point of a delay differential equation. We focus on a specific equation, a model for high speed drilling due to Stone and Askari, but the generalization to other equations should be clear. We begin by reviewing the literature on center manifolds and delay differential equations. After outlining the theoretical setting for calculating center manifolds, we show how the computations may be implemented in the symbolic algebra package Maple™. We conclude by discussing extensions of the center manifold approach as well as alternate approaches.

**Keywords:** Delay differential equation · Center manifold · Symbolic computation

### 8.1 Introduction

To begin, we briefly review some results and terminology from the theory of ordinary differential equations (ODEs). Consider an autonomous ODE

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}), \quad (8.1)$$

which admits an equilibrium point,  $\mathbf{x}^*$ . The linearization of (8.1) about  $\mathbf{x}^*$  is given by

$$\mathbf{x}' = A\mathbf{x}, \quad (8.2)$$

where  $A = D\mathbf{f}(\mathbf{x}^*)$ . Recall that  $\mathbf{x}^*$  is called *nonhyperbolic* if at least one of the eigenvalues of  $A$  has zero real part. Given a complete set of generalized eigenvectors for the eigenvalues of  $A$  with zero real part, one can construct a basis for the subspace solutions of (8.2) corresponding to these eigenvalues. This subspace is called the *center eigenspace* of (8.2). Nonhyperbolic equilibrium points are important as they often occur at bifurcation points of a differential equation. The *center manifold*

is a powerful tool for studying the behavior of solutions (and hence the nature of the bifurcation) of (8.1) in a neighborhood of a nonhyperbolic equilibrium point. It is a nonlinear manifold, which is tangent to the center eigenspace at  $\mathbf{x}^*$ . For a more detailed review of the theory and construction of center manifolds for ODEs see [17, Sect. 3.2], [49, Sect. 2.1] or [33, Sect. 2.12].

In this eighth chapter, we study the center manifolds for nonhyperbolic equilibrium points of delay differential equations (DDEs). In general, one cannot find the center manifold exactly, thus one must construct an approximation. Some authors have performed the construction by hand, e.g., [11, 12, 14, 18–20, 23–25, 28, 30–32, 38, 46, 47, 51, 53]. However, the construction generally involves a lot of computation and is most easily accomplished either numerically or with the aid of a symbolic algebra package. Here we focus on the symbolic algebra approach, which has been used by several authors, e.g., [3, 7, 34–36, 45, 48, 54–56]. Unfortunately, there is rarely space in journal articles to give details of the implementation of such computations. Thus, the purpose here is to give these details, for a particular example DDE and for the symbolic algebra package Maple™, so that other authors may reproduce them in other contexts.

In the following section, we will outline the theoretical setting for calculating center manifolds similar to the discussion of the sixth chapter. In the second section, we will show how the computations may be implemented in the symbolic algebra package Maple™ by applying the theory to a model due to Stone and Askari [44]. In the final section, we will discuss extensions of this approach as well as alternate approaches.

## 8.2 Theory

In this section, we briefly outline the theoretical setting for calculating center manifolds. More detail on the theory can be found in [2, 15, 22].

Consider the general delay differential equation

$$\mathbf{x}'(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{x}(t - \tau_1), \dots, \mathbf{x}(t - \tau_p); \mu), \quad (8.3)$$

where  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^n \times \dots \times \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^n$ ,  $p$  is a positive integer and  $\mu \in \mathbb{R}^k$  and  $\tau_j > 0$ ,  $j = 1, \dots, p$  are parameters of the model. We shall assume that  $\mathbf{g}$  is as smooth as necessary for our subsequent computations (i.e.,  $\mathbf{g} \in C^r$  for  $r$  large enough) and the equation admits an equilibrium solution  $\mathbf{x}(t) = \mathbf{x}^*$ . In general  $\mathbf{x}^*$  may depend on  $\mu$ , but not on the  $\tau_j$ . Shifting the equilibrium to zero and separating the linear and nonlinear terms gives

$$\mathbf{x}'(t) = A_0(\mu) \mathbf{x}(t) + \sum_{j=1}^p A_j(\mu) \mathbf{x}(t - \tau_j) + \mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \tau_1), \dots, \mathbf{x}(t - \tau_p); \mu), \quad (8.4)$$

where

$$A_j(\mu) = D_{j+1} \mathbf{g}(\mathbf{x}^*, \dots, \mathbf{x}^*; \mu), \quad (8.5)$$

and

$$\mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \tau_1), \dots, \mathbf{x}(t - \tau_p); \mu) = \mathbf{g}(\mathbf{x}(t), \mathbf{x}(t - \tau_1), \dots, \mathbf{x}(t - \tau_p); \mu) - A_0(\mu) \mathbf{x}(t) - \sum_{j=1}^p A_j(\mu) \mathbf{x}(t - \tau_j). \quad (8.6)$$

Here  $D_j \mathbf{g}$  means the Jacobian of  $\mathbf{g}$  with respect to its  $j$ th argument.

Let  $\tau = \max_j \tau_j$ . To pose an initial value problem at  $t = t_0$  for this DDE, one must specify the value of  $\mathbf{x}(t)$  not just at  $t = t_0$ , but on the whole interval  $[t_0 - \tau, t_0]$ . Thus an appropriate initial condition is

$$\mathbf{x}(t_0 + \theta) = \zeta_0(\theta), \quad -\tau \leq \theta \leq 0, \quad (8.7)$$

where  $\zeta_0 : [-\tau, 0] \rightarrow \mathbb{R}^n$  is a given function. It can be shown (see e.g. [22, Sect. 2.2]) that if  $\zeta_0$  is continuous and  $\mathbf{f}$  is Lipschitz there exists a unique solution to the initial value problem (8.4)–(8.7), which is defined and continuous on a (maximal) interval  $[t_0 - \tau, \beta)$ ,  $\beta > 0$ . In the following, we will assume that  $\zeta_0$  and  $\mathbf{f}$  satisfy these conditions.

To define an appropriate phase space for the solutions of the DDE, make the following definition

$$\mathbf{x}_t(\theta) \stackrel{\text{def}}{=} \mathbf{x}(t + \theta), \quad -\tau \leq \theta \leq 0.$$

Note that the initial condition can now be expressed as  $\mathbf{x}_{t_0} = \zeta_0$  and that  $\mathbf{x}_t$  will be a continuous mapping from  $[-\tau, 0] \rightarrow \mathbb{R}^n$  for each  $t \in [t_0, \beta)$ .

With this in mind, it is usual [22] to take the phase space for (8.4) to be the Banach space  $\mathcal{C} \stackrel{\text{def}}{=} C([-\tau, 0], \mathbb{R}^n)$  of continuous mappings from  $[-\tau, 0]$  into  $\mathbb{R}^n$ , equipped with the norm

$$\|\zeta\|_\tau = \sup_{\theta \in [-\tau, 0]} \|\zeta(\theta)\|,$$

where  $\|\cdot\|$  is the usual Euclidean norm on  $\mathbb{R}^n$ . We can then define the flow for the DDE as a mapping on  $\mathcal{C}$ , which takes the initial function  $\zeta_0$  into the function  $\mathbf{x}_t$ .

The equation (8.4) for  $\mathbf{x}(t)$  can be expressed as a *functional differential equation* (FDE)

$$\mathbf{x}'(t) = L(\mathbf{x}_t; \mu) + \mathbf{F}(\mathbf{x}_t; \mu), \quad (8.8)$$

where  $L : \mathcal{C} \times \mathbb{R}^k \rightarrow \mathbb{R}^n$  is a linear mapping defined by

$$L(\phi; \mu) = A_0(\mu) \phi(0) + \sum_{j=1}^p A_j(\mu) \phi(-\tau_j), \quad (8.9)$$

and  $\mathbf{F} : \mathcal{C} \times \mathbb{R}^k \rightarrow \mathbb{R}^n$  is a nonlinear functional defined by

$$\mathbf{F}(\phi; \mu) = \mathbf{f}(\phi(0), \phi(-\tau_1), \dots, \phi(-\tau_p); \mu). \quad (8.10)$$

As shown in, e.g., [11, 12, 51] one may extend (8.8) to a differential equation for  $\mathbf{x}_t(\theta)$  as follows

$$\frac{d}{dt}\mathbf{x}_t(\theta) = \begin{cases} \frac{d}{d\theta}(\mathbf{x}_t(\theta)) & , \quad -\tau \leq \theta < 0 \\ L(\mathbf{x}_t; \mu) + \mathbf{F}(\mathbf{x}_t; \mu) & , \quad \theta = 0 \end{cases} . \quad (8.11)$$

This equation will be important for the center manifold construction.

### 8.2.1 Linearization

Clearly (8.4) and (8.8) admit the trivial solution  $\mathbf{x}(t) = 0$ ,  $\forall t$ , which corresponds to the equilibrium solution  $\mathbf{x}(t) = \mathbf{x}^*$  of (8.3). The stability of this equilibrium solution can be studied via the linearization of (8.8) about the trivial solution:

$$\mathbf{x}'(t) = L(\mathbf{x}_t; \mu) \quad (8.12)$$

or, in the DDE form,

$$\mathbf{x}'(t) = A_0(\mu)\mathbf{x}(t) + \sum_{j=1}^p A_j(\mu)\mathbf{x}(t - \tau_j). \quad (8.13)$$

Substituting the ansatz  $\mathbf{x}(t) = e^{\lambda t}\mathbf{v}$ ,  $\mathbf{v} \in \mathbb{R}^n$  into (8.13) yields the matrix vector equation

$$\left[ \lambda I - A_0(\mu) - \sum_{j=1}^p A_j(\mu)e^{-\lambda \tau_j} \right] \mathbf{v} = 0, \quad (8.14)$$

which we will sometimes write in the compact form  $\Delta(\lambda; \mu)\mathbf{v} = 0$ . Requiring non-trivial solutions ( $\mathbf{v} \neq \mathbf{0}$ ) yields the constraint  $\det(\Delta(\lambda; \mu)) = 0$ , i.e., that  $\lambda$  is a root of the characteristic equation

$$\det \left[ \lambda I - A_0(\mu) - \sum_{j=1}^p A_j(\mu)e^{-\lambda \tau_j} \right] = 0. \quad (8.15)$$

It can be shown [22, Corollary 7.6.1] that the trivial solution of (8.12) (or (8.13)) will be asymptotically stable (and hence the equilibrium solution of (8.33) will be locally asymptotically stable) if all the roots of (8.15) have negative real parts. We will call these roots the eigenvalues of the equilibrium point.

Consider a point,  $\mu = \mu_c$ , in the parameter space where the characteristic equation (8.15) has  $m$  roots with zero real parts and the rest of the roots have negative real parts. The following results are shown in [22]. At such a point there exists a decomposition of the solution space for the linear FDE (8.12) as  $\mathcal{C} = N \oplus S$ , where  $N$  is an  $m$ -dimensional subspace spanned by the solutions to (8.12) corresponding to the eigenvalues with zero real part,  $S$  is infinite dimensional and  $N$  and  $S$  are invariant under the flow associated with (8.12).  $N$  and  $S$  are analogous to the center and stable eigenspaces for ODEs.

For simplicity, we will assume that all the eigenvalues with zero real part have multiplicity one. This includes the most common cases studied: single Hopf bifurcation, double Hopf bifurcation (with nonidentical frequencies) and zero-Hopf bifurcation. For a discussion of DDEs with a zero eigenvalue of multiplicity two (Bogdanov Takens bifurcation) or three, see [6] or [36]. For a discussion of DDEs where complex conjugate eigenvalues with higher multiplicity arise due to symmetry, see [7, 18–20, 26, 31, 52–54]. For a general discussion of eigenspaces associated with eigenvalues of higher multiplicity in DDEs see [22, Sect. 7.4].

Let  $\{\phi_1(t), \phi_2(t), \dots, \phi_m(t)\}$  be a basis for  $N$  and  $\{\lambda_1, \lambda_2, \dots, \lambda_m\}$  the corresponding eigenvalues. This basis can be constructed in a similar manner to that for ODEs. Since  $\text{Re}(\lambda_k) = 0$  for each  $k$ , either  $\lambda_k = 0$  or  $\lambda_k = i\omega_k$ . In the latter case, it is easy to check that  $-i\omega_k$  is also a root, and we will order the eigenvalues so that  $\lambda_{k+1} = -i\omega_k$ . With the restriction of simple eigenvalues, the construction of the basis functions is straight forward. If  $\lambda_k = 0$ , then  $\phi_k = \mathbf{v}_k$  where  $\mathbf{v}_k$  a solution of  $\Delta(0; \mu_c)\mathbf{v}_k = 0$ . If  $\lambda_k = i\omega$ , then  $\phi_k = \text{Re}(e^{i\omega_k t}\mathbf{v}_k)$  and  $\phi_{k+1} = \text{Im}(e^{i\omega_k t}\mathbf{v}_k)$ , where  $\mathbf{v}_k$  a solution of  $\Delta(i\omega_k; \mu_c)\mathbf{v}_k = 0$ . In the following, we will usually write the basis as an  $n \times m$  matrix, with the  $k$ th column given by  $\phi_k$ , viz.:

$$\Phi(t) = [\phi_1(t) | \phi_2(t) | \dots | \phi_m(t)]. \quad (8.16)$$

A simple calculation then shows that  $\phi$  satisfies the following matrix ordinary differential equation:

$$\Phi' = \Phi B, \quad (8.17)$$

where  $B$  is a block diagonal matrix, with block  $[0]$  for each zero eigenvalue and block

$$\begin{bmatrix} 0 & \omega_k \\ -\omega_k & 0 \end{bmatrix}$$

for each pair of complex conjugate eigenvalues,  $\pm i\omega_k$ .

Note that the basis functions may also be treated as functions on  $\mathcal{C}$ , by changing their argument to  $\theta \in [-\tau, 0]$ . Now consider

$$\begin{aligned} L(e^{\lambda_k \theta} \mathbf{v}_k; \mu_c) &= A_0(\mu_c)\mathbf{v}_k + \sum_{j=1}^p A_j(\mu_c)e^{-\lambda_k \tau_j} \mathbf{v}_k \\ &= \lambda_k \mathbf{v}_k, \end{aligned}$$

which follows from (8.14). This implies that  $L(\phi_k; \mu_c) = 0$  when  $\lambda_k = 0$ , and when  $\lambda_k = i\omega_k$ ,  $L(\phi_k; \mu_c) = \omega \phi_{k+1}$  and  $L(\phi_{k+1}; \mu_c) = -\omega \phi_k$ . We then have the following result:

$$L(\Phi; \mu_c) = \Phi(0)B. \quad (8.18)$$

As for ODEs, the decomposition of the solution space may be accomplished via the introduction of the adjoint equation for (8.12). However, a different equation, which is closely related to the adjoint equation, may also be used to decompose the solution space. It turns out that this latter equation is useful for the center manifold construction, so we focus on it.

Let  $R^{n*}$  be the  $n$ -dimensional row vectors and  $\mathcal{C}^* = C([0, \tau], \mathbb{R}^{n*})$ . For  $\psi \in \mathcal{C}^*$  and  $\phi \in \mathcal{C}$ , define the following bilinear form

$$\langle \psi, \phi \rangle = \sum_{i=1}^n \psi_i(0) \phi_i(0) + \sum_{j=1}^p \int_{-\tau}^0 \psi(\sigma + \tau_j) A_j \phi(\sigma) d\sigma. \quad (8.19)$$

As shown in [22, Sect. 7.5], this can be used to define a system dual to (8.12) given by

$$\mathbf{y}'(t) = L^T(\mathbf{y}_t; \mu), \quad s \leq 0, \quad (8.20)$$

where  $y^s = y(s + \xi)$ ,  $0 \leq \xi \leq \tau$  and  $L^T$  is a linear mapping on  $\mathcal{C}^* \times \mathbb{R}^k$  given by

$$L^T(\psi; \mu) = -\psi(0) A_0(\mu) - \sum_{j=1}^p \psi(\tau_j) A_j(\mu). \quad (8.21)$$

Equation (8.20) is called the *transposed system* by [22]. In the literature, it is sometimes called the formal adjoint. The corresponding differential equation is

$$\mathbf{y}'(s) = -\mathbf{y}(s) A_0(\mu) - \sum_{j=1}^p \mathbf{y}(s + \tau_j) A_j(\mu), \quad s \leq 0. \quad (8.22)$$

Using the ansatz  $\mathbf{y}(s) = \mathbf{w} e^{-\lambda s}$ ,  $\mathbf{w} \in \mathbb{R}^{n*}$  and proceeding as for (8.12), shows that  $\mathbf{w}$  must satisfy  $\mathbf{w} \Delta(\lambda; \mu) = 0$ . Thus the characteristic equation of (8.22) is just (8.15). It follows that the trivial solutions of (8.22) and (8.13) have the same eigenvalues.

Let

$$\Psi(s) = \begin{bmatrix} \psi_1(s) \\ \vdots \\ \psi_m(s) \end{bmatrix},$$

be a basis for the solutions of (8.20) (or, equivalently, (8.22)) corresponding to the  $m$  eigenvalues with zero real part (i.e. the “center eigenspace” of (8.22)). Note that the  $\psi_j$  are row vectors and that they can be considered as functions on  $\mathcal{C}^*$  if we change their argument to  $\xi \in [0, \tau]$ . The fundamental result used in the center manifold construction is that  $\Psi$  may be used to decompose the solution space. See [22, Sect. 7.5] for details and proofs. In particular, for any  $\zeta \in S$ ,

$$\langle \psi_j, \zeta \rangle = 0, \quad j = 1, \dots, m.$$

Further, we can choose a basis so that  $\langle \Psi, \Phi \rangle = \mathbf{I}$ , where  $\langle \Psi, \Phi \rangle$  is the  $m \times m$  matrix with  $i, j$  elements  $\langle \psi_i, \phi_j \rangle$  and  $\mathbf{I}$  is the  $m \times m$  identity matrix. Thus for any  $\zeta \in N$  we have  $\zeta = \Phi \mathbf{u}$  where  $\mathbf{u} = \langle \Psi, \zeta \rangle \in \mathbb{R}^m$ . Finally, one can show that

$$\Psi' = -B\Psi \quad \text{and} \quad L^T(\Psi; \mu_c) = -B\Psi(0), \quad (8.23)$$

where  $B$  is the same block diagonal matrix as in (8.17).

### 8.2.2 Nonlinear Equation

Now let us return to the nonlinear equation (8.8). For the rest of this section we will assume that  $\mu = \mu_c$  and hence that the characteristic equation (8.15) has  $m$  eigenvalues with zero real parts and all other eigenvalues have negative real parts. In this situation [22, Chap. 10] has shown that there exists, in the solution space  $\mathcal{C}$  for the nonlinear FDE (8.8), an  $m$  dimensional center manifold. Since all the other eigenvalues have negative real parts, this manifold is attracting and the long term behavior of solutions to the nonlinear equation is well approximated by the flow on this manifold. In particular, studying the flow on this manifold will enable us to characterize the bifurcation, which occurs as a  $\mu$  passes  $\mu_c$ . Below, we outline the steps involved in computing this manifold. The approach we take follows the work of [21] and [51] (scalar case) and of [2] (vector case). Since all our computations will be done for  $\mu = \mu_c$ , we not write the dependence on  $\mu$  explicitly.

To begin, we note that points on the local center manifold of  $\mathbf{0}$  can be expressed as the sum of a linear part belonging to  $N$  and a nonlinear part belonging to  $S$ , i.e.,

$$W_{\text{loc}}^c(\mathbf{0}) = \{\phi \in \mathcal{C} \mid \phi = \Phi \mathbf{u} + \mathbf{h}(\mathbf{u})\},$$

where  $\Phi(\theta)$ ,  $\theta \in [-\tau, 0]$  is the basis for  $N$  introduced above,  $\mathbf{u} \in \mathbb{R}^m$ ,  $\mathbf{h}(\mathbf{u}) \in S$  and  $\|\mathbf{u}\|$  is sufficiently small. The solutions of (8.8) on this center manifold are then given by  $\mathbf{x}(t) = \mathbf{x}_t(0)$ , where  $\mathbf{x}_t(\theta)$  is a solution of (8.11) satisfying

$$\mathbf{x}_t(\theta) = \Phi(\theta)\mathbf{u}(t) + \mathbf{h}(\theta, \mathbf{u}(t)). \quad (8.24)$$

To find the center manifold and the solutions on it, we proceed as follows. Substituting (8.24) into (8.11) yields

$$\left[ \Phi(\theta) + \frac{\partial \mathbf{h}}{\partial \mathbf{u}} \right] \dot{\mathbf{u}}(t) = \begin{cases} \Phi'(\theta)\mathbf{u}(t) + \frac{\partial \mathbf{h}}{\partial \theta} & , \quad -\tau \leq \theta < 0 \\ L(\Phi(\theta))\mathbf{u}(t) + L(\mathbf{h}(\theta, \mathbf{u}(t))) \\ + \mathbf{F}[\Phi(\theta)\mathbf{u}(t) + \mathbf{h}(\theta, \mathbf{u}(t))] & , \quad \theta = 0. \end{cases} \quad (8.25)$$

Using (8.17) and (8.18) in (8.25) we obtain

$$\left[ \Phi(\theta) + \frac{\partial \mathbf{h}}{\partial \mathbf{u}} \right] \dot{\mathbf{u}}(t) = \begin{cases} \Phi(\theta)B\mathbf{u}(t) + \frac{\partial \mathbf{h}}{\partial \theta} & , \quad -\tau \leq \theta < 0, \\ \Phi(0)B\mathbf{u}(t) + L(\mathbf{h}(\theta, \mathbf{u}(t))) \\ + \mathbf{F}[\Phi(\theta)\mathbf{u}(t) + \mathbf{h}(\theta, \mathbf{u}(t))] & , \quad \theta = 0. \end{cases} \quad (8.26)$$

This coupled system must be solved for  $\mathbf{u}(t)$  and  $\mathbf{h}(\theta, \mathbf{u}(t))$ .

To derive the equation for  $\mathbf{u}(t)$  we will use the bilinear form (8.19). First, we note some useful results. Since  $\mathbf{h}(\theta, \mathbf{u}) \in S$  for any  $\mathbf{u}$ ,

$$\langle \Psi(\xi), \mathbf{h}(\theta, \mathbf{u}(t)) \rangle = 0.$$

It then follows from the definition of the partial derivative that

$$\langle \Psi(\xi), \frac{\partial \mathbf{h}}{\partial \mathbf{u}}(\theta, \mathbf{u}(t)) \rangle = 0.$$

Finally, using (8.23) we have

$$\begin{aligned} & \Psi(0)L(\mathbf{h}(\theta, \mathbf{u})) + \sum_{j=1}^p \int_{-\tau}^0 \Psi(\sigma + \tau_j) A_j \frac{\partial \mathbf{h}}{\partial \sigma} d\sigma \\ &= -L^T(\Psi(\xi))\mathbf{h}(0, \mathbf{u}(t)) - \sum_{j=1}^p \int_{-\tau}^0 \Psi'(\sigma + \tau_j) A_j \mathbf{h}(\sigma, \mathbf{u}(t)) d\sigma \\ &= B\Psi(0)\mathbf{h}(0, \mathbf{u}(t)) + \sum_{j=1}^p \int_{-\tau}^0 B\Psi(\sigma + \tau_j) A_j \mathbf{h}(\sigma, \mathbf{u}(t)) d\sigma \\ &= B\langle \Psi(\xi), \mathbf{h}(\theta, \mathbf{u}) \rangle \\ &= 0. \end{aligned}$$

Applying the bilinear form to  $\Psi$  and (8.26) and using these results gives the following system of ODEs for  $\mathbf{u}(t)$ :

$$\dot{\mathbf{u}}(t) = B\mathbf{u}(t) + \Psi(0)\mathbf{F}[\Phi(\theta)\mathbf{u}(t) + \mathbf{h}(\theta, \mathbf{u}(t))] . \quad (8.27)$$

Using (8.27) in (8.26) then yields the following system of partial differential equations for  $\mathbf{h}(\theta, \mathbf{u})$ :

$$\begin{aligned} & \frac{\partial \mathbf{h}}{\partial \mathbf{u}} \{ B\mathbf{u} + \Psi(0)\mathbf{F}[\Phi(\theta)\mathbf{u} + \mathbf{h}(\theta, \mathbf{u})] \} + \Phi(\theta)\Psi(0)\mathbf{F}[\Phi(\theta)\mathbf{u} + \mathbf{h}(\theta, \mathbf{u})] \\ &= \begin{cases} \frac{\partial \mathbf{h}}{\partial \theta} , & -\tau \leq \theta < 0 \\ L(\mathbf{h}(\theta, \mathbf{u})) + \mathbf{F}[\Phi(\theta)\mathbf{u} + \mathbf{h}(\theta, \mathbf{u})] , & \theta = 0 . \end{cases} \quad (8.28) \end{aligned}$$

Thus, the evolution of solutions on the center manifold is determined by solving (8.28) for  $\mathbf{h}(\theta, \mathbf{u})$  and then (8.27) for  $\mathbf{u}(t)$ . To solve (8.28), one uses a standard approach in center manifold theory, namely, one assumes that  $\mathbf{h}(\theta, \mathbf{u})$  may be expanded in power series in  $\mathbf{u}$ :

$$\mathbf{h}(\theta, \mathbf{u}) = \mathbf{h}_2(\theta, \mathbf{u}) + \mathbf{h}_3(\theta, \mathbf{u}) + \cdots , \quad (8.29)$$

where

$$\mathbf{h}_2(\theta, \mathbf{u}) = \begin{bmatrix} h_{11}^1(\theta)u_1^2 + \cdots + h_{1m}^1(\theta)u_1u_m + h_{22}^1(\theta)u_2^2 + \cdots + h_{mm}^1(\theta)u_m^2 \\ \vdots \\ h_{11}^n(\theta)u_1^2 + \cdots + h_{1m}^n(\theta)u_1u_m + h_{22}^n(\theta)u_2^2 + \cdots + h_{mm}^n(\theta)u_m^2 \end{bmatrix} ,$$

and similarly for  $\mathbf{h}_3$  and the higher order terms.

Before proceeding to solve for  $\mathbf{h}$ , we would like to note that to determine the terms of (8.27) to  $O(\|\mathbf{u}(t)\|^1)$ , one only needs the terms which are  $O(\|\mathbf{u}(t)\|^{1-1})$  in the series for  $\mathbf{h}$ . To see this, write  $\mathbf{F}$  in series form

$$\mathbf{F} = \mathbf{F}_2 + \mathbf{F}_3 + \cdots \quad (8.30)$$

and hence rewrite (8.27) as

$$\begin{aligned} \dot{\mathbf{u}} = & B\mathbf{u} + \Psi(0) \{ \mathbf{F}_2[\boldsymbol{\Phi}(\theta)\mathbf{u} + \mathbf{h}_2(\theta, \mathbf{u}) + \mathbf{h}_3(\theta, \mathbf{u}) + O(\|\mathbf{u}\|^4)] \\ & + \mathbf{F}_3[\boldsymbol{\Phi}(\theta)\mathbf{u} + \mathbf{h}_2(\theta, \mathbf{u}) + \mathbf{h}_3(\theta, \mathbf{u}) + O(\|\mathbf{u}\|^4)] + O(\|\mathbf{u}\|^4) \}. \end{aligned}$$

Expanding each  $\mathbf{F}_j$  in a Taylor series about  $\boldsymbol{\phi}(\theta)\mathbf{u}$  yields

$$\dot{\mathbf{u}} = B\mathbf{u} + \Psi(0) [\mathbf{F}_2(\boldsymbol{\Phi}(\theta)\mathbf{u}) + D\mathbf{F}_2(\boldsymbol{\Phi}(\theta)\mathbf{u})\mathbf{h}_2(\theta, \mathbf{u}) + \mathbf{F}_3(\boldsymbol{\Phi}(\theta)\mathbf{u})] + O(\|\mathbf{u}\|^4).$$

Thus we see that  $\mathbf{h}_2$  is only needed to calculate the third-order terms not the second-order terms. A similar result holds for the higher order terms. Of particular note is the fact that if the lowest order terms, we need in the center manifold are the same as the lowest order terms in  $\mathbf{F}$ , then there is no need to calculate  $\mathbf{h}$  at all! This is the case for a Hopf bifurcation when  $\mathbf{F}_2 = 0$ . Examples of this can be found in [27, 43, 51]. This is also the case when the normal form for a particular bifurcation is determined at second order, such as for the Bogdanov-Takens bifurcation (see, e.g., [6]) or a double Hopf bifurcation with 1:2 resonance [4].

Now let us return to solving (8.28). Substituting (8.29) and (8.30) into the first part of (8.28) and expanding the  $F_j$  about  $\boldsymbol{\Phi}(\theta)\mathbf{u}$  yields

$$\frac{\partial \mathbf{h}_2}{\partial \theta} + O(\|\mathbf{u}\|^3) = \frac{\partial \mathbf{h}_2}{\partial \mathbf{u}}(\theta, \mathbf{u})B\mathbf{u} + \boldsymbol{\Phi}(\theta)\Psi(0)\mathbf{F}_2(\boldsymbol{\Phi}(\theta)\mathbf{u}) + O(\|\mathbf{u}\|^3). \quad (8.31)$$

Equating terms with like powers of  $u_1, \dots, u_m$  in this equation yields a system of ODEs for the  $h_{jk}^i(\theta)$ . The system is linear and is easily solved to find the general solutions for the  $h_{jk}^i(\theta)$  in terms of arbitrary constants.

These arbitrary constants may be determined as follows. Substituting (8.29) and (8.30) into the second part of (8.28) and expanding the  $F_j$  about  $\boldsymbol{\phi}(\theta)\mathbf{u}$  yields

$$\begin{aligned} \left. \frac{\partial \mathbf{h}_2}{\partial \mathbf{u}} \right|_{\theta=0} & B\mathbf{u} + \boldsymbol{\Phi}(0)\Psi(0)\mathbf{F}_2(\boldsymbol{\Phi}(\theta)\mathbf{u}) + O(\|\mathbf{u}\|^3) \\ & = L(\mathbf{h}_2(\theta, \mathbf{u})) + \mathbf{F}_2(\boldsymbol{\Phi}(\theta)\mathbf{u}) + O(\|\mathbf{u}\|^3) \end{aligned} \quad (8.32)$$

Equating terms with like powers of  $u_1, \dots, u_m$  in this equation yields a set of boundary conditions for the arbitrary constants.

Once one has determined  $\mathbf{h}_2$  one may proceed to the next order of approximation and calculate  $\mathbf{h}_3$ . As discussed above, however, for most applications this is unnecessary.

### 8.3 Application

Now consider the model of [44]:

$$\eta'' + \delta\eta' + \eta - \beta(1 - \mu(\eta - \eta(t - \tau)))(p_0 + p_1\eta' + p_2\eta'^2) = 0, \quad (8.33)$$

which was developed to study the vibrations in drilling. This model is in dimensionless form, the model in physical variables can be found in [44] or [45]. The variable  $\eta$  corresponds to the amplitude of the vibrations and  $'$  to derivative with respect to time. The parameter  $1/\tau$  is proportional to the speed of rotation of the drill and  $\beta$  to the width of cut. Since these two parameters can be varied in practice, [45] chose these as the bifurcation parameters. The other parameters,  $\mu$ ,  $p_0$ ,  $p_1$ ,  $p_2$ , can be related to other physical parameters [45].

Note that this equation has an equilibrium solution  $\eta(t) = \beta p_0$ , which corresponds to the steady cutting solution. The drilling process may exhibit *chatter*, which is a self excited oscillation of the drill. The emergence of chatter in the physical system corresponds to a Hopf bifurcation in the model (8.33). In [45] the criticality of this bifurcation was studied using the center manifold construction described in Sect. 1. We will reproduce the essence of the analysis here, including the relevant commands in Maple<sup>TM</sup>11<sup>1</sup> used to perform the computations symbolically. Commands will be written in typewriter font and preceded by a `>`. Each command will be followed by the output produced when it is executed. If a command ends with a colon then no output is printed. More information on Maple can be found in the manual [29].

Shifting the equilibrium to the origin and rewriting the equation as a first order vector equation puts it in the form (8.4):

$$\mathbf{x}'(t) = A_0\mathbf{x}(t) + A_1\mathbf{x}(t - \tau) + \mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \tau)), \quad (8.34)$$

where

$$\mathbf{x}(t) = \begin{bmatrix} x(t) \\ x'(t) \end{bmatrix} = \begin{bmatrix} \eta(t) - \beta p_0 \\ \eta'(t) \end{bmatrix}, \quad (8.35)$$

$$A_0 = \begin{bmatrix} 0 & 1 \\ -1 - \beta p_0 \mu & \beta p_1 - \delta \end{bmatrix}, A_1 = \begin{bmatrix} 0 & 0 \\ \beta p_0 \mu & 0 \end{bmatrix}, \quad (8.36)$$

and

$$\mathbf{f} = \begin{bmatrix} 0 \\ \beta p_2 x'(t)^2 - \beta \mu(x(t) - x(t - \tau))(p_1 x'(t) + p_2 x'(t)^2) \end{bmatrix}. \quad (8.37)$$

Thus, in this example, the linear mapping of (8.8) is given by

$$L(\phi(\theta)) = A_0\phi(0) + A_1\phi(-\tau), \quad (8.38)$$

and  $\mathbf{F} : \mathcal{C} \rightarrow \mathbb{R}^2$  is a nonlinear functional defined by

$$\mathbf{F}(\phi(\theta)) = \mathbf{f}(\phi(0), \phi(-\tau)). \quad (8.39)$$

---

<sup>1</sup> The commands used are backward compatible to at least Maple<sup>TM</sup>9.5.

The characteristic matrix and equation for this example can be defined in Maple as follows.

```
> A0:=matrix(2,2,[[0,1],[-1-beta*p0*mu,
    beta*p1-delta]]);
```

$$A_0 := \begin{bmatrix} 0 & 1 \\ -1 - \beta p_0 \mu & \beta p_1 - \delta \end{bmatrix}$$

```
> A1:=matrix(2,2,[[0,0],[beta*p0*mu,0]]);
```

$$A_1 := \begin{bmatrix} 0 & 0 \\ \beta p_0 \mu & 0 \end{bmatrix}$$

```
> ident:=evalm(array(1..2,1..2,identity));
```

$$ident := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

```
> Delta:=evalm(lambda*ident-A0-exp(-lambda*tau)*A1);
```

$$\Delta := \begin{bmatrix} \lambda & -1 \\ 1 + \beta p_0 \mu - \beta p_0 \mu e^{-\lambda \tau} & \lambda - \beta p_1 + \delta \end{bmatrix}$$

```
> char_eq:=collect(det(Delta),lambda);
```

$$char\_eq := \lambda^2 + (-\beta p_1 + \delta)\lambda + 1 + \beta p_0 \mu - e^{(-\lambda \tau)} \beta p_0 \mu$$

The work of [45] described curves, in the  $\tau, \beta$  parameter space, along which the equilibrium solution of (8.33) loses stability. At each point on these curves, the characteristic (8.15) has a pair of pure imaginary roots and the rest of the roots have negative real parts. Equations describing where the characteristic equation has pure imaginary roots can be easily found in Maple:

```
> eq_im:=evalc(subs(lambda=I*omega,char_eq)):
eq_Re:=coeff(eq_im,I,0);
eq_Im:=coeff(eq_im,I,1);
```

$$eq\_Re := -\omega^2 + 1 + \beta p_0 \mu - \cos(\omega \tau) \beta p_0 \mu$$

$$eq\_Im := -\omega \beta p_1 + \omega \delta + \sin(\omega \tau) \beta p_0 \mu$$

[45] solved these equations to find expressions for  $\tau$  and  $\beta$  in terms of  $\omega$  and the other parameters. For fixed values of the other parameters, these expressions determined curves in the  $\tau, \beta$  parameter space, which are parametrized by  $\omega$ .

It is straightforward to check that the FDE (8.33) satisfies the conditions for a Hopf bifurcation to occur as one passes through a point on these curves (see [22, pp. 331–333] or [14, Sect. 8.2] for a statement of the Hopf bifurcation Theorem

for FDE's). To determine the criticality of this Hopf bifurcation, we compute the center manifold of the equilibrium point at the Hopf bifurcation, following the steps outlined in Sect. 8.2.

To begin, we calculate a basis for the “center eigenspace”,  $N$ . To do this we need to find the eigenfunctions corresponding to the eigenvalues  $\pm i\omega$ . In Maple this may be done as follows. First, solve  $\Delta(i\omega)\mathbf{v} = 0$ , where  $\Delta(i\omega)$  is the characteristic matrix with  $\lambda = i\omega$ .

```
> v:=matrix([ [v1], [v2] ]);
Dv:=subs(lambda=I*omega, evalm(multiply(Delta,v)));
v2res:=v2=solve(Dv[1,1], v2);
```

$$\mathbf{v} := \begin{bmatrix} v1 \\ v2 \end{bmatrix}$$

$$Dv := \begin{bmatrix} I\omega v1 - v2 \\ (1 + \beta p0\mu - e^{I\omega\tau}\beta p0\mu)v1 + (I\omega - \beta p1 + \delta)v2 \end{bmatrix}$$

$$v2res := v2 = I\omega v1.$$

Then define the complex eigenfunction and take the real and imaginary parts.

```
> yy:=map(evalc, subs(v2res, v1=1,
    evalm(exp(I*omega*theta)*v))):
Phi:=array(1..2, 1..2, [[coeff(yy[1,1], I, 0),
    coeff(yy[1,1], I, 1)],
    [coeff(yy[2,1], I, 0), coeff(yy[2,1], I, 1)]]);
```

$$\Phi := \begin{bmatrix} \cos(\omega\theta) & \sin(\omega\theta) \\ -\sin(\omega\theta)\omega & \cos(\omega\theta)\omega \end{bmatrix}$$

Similarly we define  $\mathbf{u}$  and  $\Phi\mathbf{u}$ .

```
u:=matrix([ [u1], [u2] ]);
```

$$\mathbf{u} := \begin{bmatrix} u1 \\ u2 \end{bmatrix}$$

```
Phiu:=multiply(Phi,u);
```

$$Phiu := \begin{bmatrix} \cos(\omega\theta)u1 + \sin(\omega\theta)u2 \\ -\sin(\omega\theta)\omega u1 + \cos(\omega\theta)\omega u2 \end{bmatrix}$$

Next define the matrix  $B$ .

```
> B:=matrix([ [0, omega], [-omega, 0] ]);
```

$$B := \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}.$$

We need to define the basis,  $\Psi(\xi)$ ,  $\xi \in [0, \tau]$ , for the “center eigenspace” of the transpose system. First, we calculate a general basis  $\Psi_g$  in the same way as we set up the basis  $\Phi$ .

```
> w:=array(1..2):
wD:=subs(lambda=I*omega,multiply(w,Delta)):
wlres:=w[1]=solve(wD[2],w[1]):
yy:=map(evalc,subs(wlres,w[2]=1,lambda=I*omega,
    evalm(w*exp(-lambda*xi)))):
Psi_g:=array(1..2,1..2,[[coeff(yy[1],I,0),
    coeff(yy[2],I,0)],
    [coeff(yy[1],I,1),coeff(yy[2],I,1)]]);
```

$$\Psi_g := \begin{bmatrix} \cos(\omega\xi)(-\beta p_1 + \delta) + \sin(\omega\xi)\omega & \cos(\omega\xi) \\ -\sin(\omega\xi)(-\beta p_1 + \delta) + \cos(\omega\xi)\omega & -\sin(\omega\xi) \end{bmatrix}$$

We now wish to find a basis  $\Psi$  such that  $\langle \Psi, \Phi \rangle = \mathbf{I}$ . The elements of  $\Psi$  will be linear combinations of those of  $\Psi_g$ , i.e.,  $\Psi = K\Psi_g$ , where  $K$  is a  $2 \times 2$  matrix of constants. Thus we have

$$\begin{aligned} \mathbf{I} &= \langle \Psi, \Phi \rangle \\ &= \langle K\Psi_g, \Phi \rangle \\ &= K\langle \Psi_g, \Phi \rangle, \end{aligned}$$

Which implies that  $K = \langle \Psi_g, \Phi \rangle^{-1}$ .

For this example the bilinear form (8.19) becomes

$$\begin{aligned} \langle \psi, \phi \rangle &= \sum_{j=1}^2 \psi_j(0)\phi_j(0) + \int_{-\tau}^0 \psi(\sigma + \tau)A_1\phi(\sigma) d\sigma \\ &= \psi(0)\phi(0) + \beta\mu p_0 \int_{-\tau}^0 \psi_2(\sigma + \tau)\phi_1(\sigma) d\sigma. \end{aligned}$$

We define this bilinear form as a procedure as follows:

```
> bilinear_form:=proc(rowv,colv)
local pstep;
pstep:=subs(xi=0,theta=0,innerprod(rowv,colv))
+int(subs(xi=sigma+tau,theta=sigma,
innerprod(rowv,A1,colv)),sigma=-tau .. 0);
RETURN(pstep)
end;
```

Note that the command `innerprod` calculates the dot product when given two vectors and the vector-matrix-vector product when given two vectors and a matrix.

Next we apply the bilinear form to each row of  $\Psi_g$  and each column of  $\Phi$  and store the result in the matrix `produit`. We then invert `produit` and multiply the result by  $\Psi_g$ .<sup>2</sup>

```
> rowvec:=array(1..2): colvec:=array(1..2):
    produit:=array(1..2):
> for I1 from 1 to 2 do
    for I2 from 1 to 2 do
        rowvec:=row(Psi_g, I1);
        colvec:=col(Phi, I2);
        produit[I1, I2]:=eval(bilinear_form
                               (rowvec, colvec));
    od;
od;
> K:=inverse(produit):
> PPSi:=map(simplify, multiply(K, Psi_g)):
```

In fact, all we need for subsequent calculations is  $\Psi(0)$ . To keep the expressions from getting too large, we will define an empty matrix  $\Psi_0$  to use as a place holder. We will store the actual values of  $\Psi(0)$  in the list `Psi0_vals`.

```
Psi0:=matrix(2, 2);
Psi0_res:=map(simplify, map(eval,
    subs(xi=0, evalm(PPSi)))):
Psi0_vals:=[Psi0[1,1]=Psi0_res[1,1],
    Psi0[1,2]=Psi0_res[1,2],
    Psi0[2,1]=Psi0_res[2,1],
    Psi0[2,2]=Psi0_res[2,2]]:
```

$$\Psi_0 = \text{array}(1..2, 1..2, []).$$

Now, to determine the criticality of the Hopf bifurcation, one need only find the terms up to and including those which are  $O(\|\mathbf{u}(t)\|^3)$  in (8.27). Thus, as discussed in the previous section, we only need the quadratic terms in the series for  $\mathbf{h}$ . We thus define

```
h:=matrix([ [h1_11(theta)*u1^2+h1_12(theta)*u1*u2
    +h1_22(theta)*u2^2],
    [h2_11(theta)*u1^2+h2_12(theta)*u1*u2+h2_22
    (theta)*u2^2]]);
```

$$h := \begin{bmatrix} h_{1,11}(\theta)u_1^2 + h_{1,12}(\theta)u_1u_2 + h_{1,22}(\theta)u_2^2 \\ h_{2,11}(\theta)u_1^2 + h_{2,12}(\theta)u_1u_2 + h_{2,22}(\theta)u_2^2 \end{bmatrix} \quad (8.40)$$

---

<sup>2</sup> Note that `Psi` is a reserved word, so we use `PPSi` instead.

We define the linear and nonlinear parts of the DE as follows

```
> x:=matrix([[x1],[x2]]);
xt:=matrix([[x1t],[x2t]]);
lin:=evalm(multiply(A0,x)+multiply(A1,xt));
f:= beta*p2*x2^2-beta*mu*p1*x1*x2+beta*mu*p1*x1t*x2
-beta*mu*p2*x1*x2^2+beta*mu*p2*x1t*x2^2:
nlin:=matrix([[0],[f]]);
```

$$x := \begin{bmatrix} x1 \\ x2 \end{bmatrix}$$

$$xt := \begin{bmatrix} x1t \\ x2t \end{bmatrix}$$

$$lin := \begin{bmatrix} x2 \\ (-1 - \beta \mu p0)x1 + (\beta p1 - \delta)x2 + \beta \mu p0x1t \end{bmatrix}$$

$$nlin := \begin{bmatrix} 0 \\ \beta p2x2^2 - \beta \mu p1x1x2 + \beta \mu p1x1tx2 - \beta \mu p2x1x2^2 + \beta \mu p2x1tx2^2 \end{bmatrix}.$$

Then we define the expressions, in terms of the coordinates **u**, for points on the centre eigenspace, **x\_ce**, and on the centre manifold, **x\_cm**.

```
> Phiu0:=map(eval,subs(theta=0,evalm(Phiu))):
Phiut:=map(eval,subs(theta=-tau,evalm(Phiu))):
x_ce:=[x1=Phiu0[1,1],x2=Phiu0[2,1],x1t=Phiut[1,1],
x2t=Phiut[2,1]];
Phiuh0:=map(eval,subs(theta=0,evalm(Phiu+h))):
Phiuht:=map(eval,subs(theta=-tau,evalm(Phiu+h))):
x_cm:=[x1=Phiuh0[1,1],x2=Phiuh0[2,1],
x1t=Phiuht[1,1],x2t=Phiuht[2,1]];
```

$$x_{ce} := [x1 = u1, x2 = \omega u2, x1t = \cos(\omega \tau)u1 - \sin(\omega \tau)u2, \\ x2t = \sin(\omega \tau)\omega u1 + \cos(\omega \tau)\omega u2]$$

$$x_{cm} := [x1 = u1 + h1.11(0)u1^2 + h1.12(0)u1u2 + h1.22(0)u2^2, \\ x2 = \omega u2 + h2.11(0)u1^2 + h2.12(0)u1u2 + h2.22(0)u2^2, \\ x1t = \cos(\omega \tau)u1 - \sin(\omega \tau)u2 + h1.11(-\tau)u1^2 \\ + h1.12(-\tau)u1u2 + h1.22(-\tau)u2^2, \\ x2t = \sin(\omega \tau)\omega u1 + \cos(\omega \tau)\omega u2 + h2.11(-\tau)u1^2 \\ + h2.12(-\tau)u1u2 + h2.22(-\tau)u2^2]$$

We can now define differential equations for the  $h_{jk}^i$ . First define the left hand side of (8.31).

```
> delhs:=map(diff,h,theta);
```

$$delhs := \begin{bmatrix} \left( \frac{d}{d\theta} h_{1.11}(\theta) \right) u_1^2 + \left( \frac{d}{d\theta} h_{1.12}(\theta) \right) u_1 u_2 + \left( \frac{d}{d\theta} h_{1.22}(\theta) \right) u_2^2 \\ \left( \frac{d}{d\theta} h_{2.11}(\theta) \right) u_1^2 + \left( \frac{d}{d\theta} h_{2.12}(\theta) \right) u_1 u_2 + \left( \frac{d}{d\theta} h_{2.22}(\theta) \right) u_2^2 \end{bmatrix}$$

Now define the right-hand side of (8.31).

```
> dhdu:=matrix([ [diff(h[1,1],u1), diff(h[1,1],u2)],
                  [diff(h[2,1],u1), diff(h[2,1],u2)]]);
```

$$dhdu := \begin{bmatrix} 2h_{1.11}(\theta)u_1 + h_{1.12}(\theta)u_2 & h_{1.12}(\theta)u_1 + 2h_{1.22}(\theta)u_2 \\ 2h_{2.11}(\theta)u_1 + h_{2.12}(\theta)u_2 & h_{2.12}(\theta)u_1 + 2h_{2.22}(\theta)u_2 \end{bmatrix}$$

```
> derhs:=map(collect,map(expand,evalm
                          multiply(dhdu,multiply(B,u))+
                          multiply(Phi,multiply(Psi0,
                          [0,subs(x_ce,f)]))),
              [u2,u2],distributed,factor):
```

The expression for derhs is quite long, so we do not display it. Now we put together the right-hand side and left-hand side. The coefficient of each distinct monomial,  $u_1^k u_2^j$ ,  $j+k=2$ , determines one differential equation. We display two of them as examples.

```
> hdes:=delhs-derhs:
de1:=coeff(coeff(hdes[1,1],u1^2),u2,0);
de2:=coeff(coeff(hdes[1,1],u1),u2);
de3:=coeff(coeff(hdes[1,1],u2^2),u1,0);
de4:=coeff(coeff(hdes[2,1],u1^2),u2,0);
de5:=coeff(coeff(hdes[2,1],u1),u2);
de6:=coeff(coeff(hdes[2,1],u2^2),u1,0):
```

$$de1 := \left( \frac{d}{d\theta} h_{1.11}(\theta) \right) + \omega h_{1.12}(\theta)$$

$$de2 := \left( \frac{d}{d\theta} h_{1.12}(\theta) \right) - 2\omega(h_{1.11}(\theta) - h_{1.22}(\theta)) + \omega\beta\mu p_1(1 - \cos(\omega\tau)) \\ (\cos(\omega\theta)\Psi_{1,2}(0) + \sin(\omega\theta)\Psi_{2,2}(0))$$

Now define the list of differential equations and functions to solve for.

```
> des:={de1,de2,de3,de4,de5,de6}:
fns:={coeff(h[1,1],u1^2),coeff(coeff(h[1,1],u1),u2),
      coeff(h[1,1],u2^2),coeff(h[2,1],u1^2),
      coeff(coeff(h[2,1],u1),u2),coeff(h[2,1],u2^2)};
```

$fns := \{h1\_11(\theta), h1\_12(\theta), h1\_22(\theta), h2\_11(\theta), h2\_12(\theta), h2\_22(\theta)\}$

The differential equations are linear and are easily solved to find the general solutions for the  $h_{jk}^i(\theta)$  in terms of six arbitrary constants using the command `dsolve`. For convenience, we rename the arbitrary constants.

```
> temp:=dsolve(des,fns):
  changeC:=[_C1=C1,_C2=C2,_C3=C3,_C4=C4,
            _C5=C5,_C6=C6];
  hsoln:=simplify(expand(evalc(subs(changeC,
                                     value(temp))))):
```

The solutions are quite long, so we show only one example.

```
> collect(hsoln[6],[Psi0[1,1],Psi0[1,2],Psi0[2,1],
              Psi0[2,2],p1,p2],factor);
```

$$\begin{aligned} h1\_22(\theta) = & \left( -\frac{1}{3} \beta \mu (\cos(\omega\theta) + \sin(\omega\theta) \sin(\omega\tau) - \cos(\omega\theta) \cos(\omega\tau)) p1 \right. \\ & + \frac{1}{3} \beta \sin(\omega\theta) \omega p2 \left. \right) \Psi_{0,1,2} + \left( \frac{1}{3} \beta \mu (\cos(\omega\theta) \sin(\omega\tau) \right. \\ & - \sin(\omega\theta) + \sin(\omega\theta) \cos(\omega\tau)) p1 - \frac{1}{3} \beta \cos(\omega\theta) \omega p2 \left. \right) \Psi_{0,2,2} \\ & - C6 \cos(\omega\theta)^2 + C2 + \frac{1}{2} C6 + C5 \sin(\omega\theta) \cos(\omega\theta) \end{aligned}$$

Recall that the values for  $\Psi_{0,i,j}$  are stored in the list `Psi0_vals`. Later we will need the values of  $h_{jk}^i(0)$  and  $h_{jk}^i(-\tau)$ , so we store them in the sets `hsoln0` and `hsolnt`.

```
> hsoln0:=simplify(eval(subs(theta=0,hsoln))):
  hsolnt:=simplify(eval(subs(theta=-tau,hsoln))):
```

We now set up the boundary conditions to solve for the arbitrary constants,  $C1, C2, \dots$ . Note that the left-hand side of (8.32) is just the right-hand side of (8.31) with  $\theta = 0$ .

```
> bclhs:=map(eval,subs(simpres,theta=0,evalm(derhs))):
  bcrhs:=map(collect,evalm(subs(x_cm,evalm(lin))+
                               subs(x_ce,evalm(nonlin))),[u1,u2]);
```

Now we put together the right-hand side and left-hand side. The coefficient of each distinct monomial,  $u1^k u2^j$ ,  $j+k=2$ , determines one boundary condition.

```
> consts:=[C1,C2,C3,C4,C5,C6];
  bceq:=subs(hsoln0,hsolnt,evalm(bclhs-bcrhs)):
  bc1:=collect(coeff(coeff(bceq[1,1],u1,2),u2,0),
               consts);
```

```

bc2:=collect (coeff (coeff (bceq[1,1],u1,1),u2,1),
               consts);
bc3:=collect (coeff (coeff (bceq[1,1],u1,0),u2,2),
               consts);
bc4:=collect (coeff (coeff (bceq[2,1],u1,2),u2,0),
               consts);
bc5:=collect (coeff (coeff (bceq[2,1],u1,1),u2,1),
               consts);
bc6:=collect (coeff (coeff (bceq[2,1],u1,0),u2,2),
               consts);

```

Form the list of boundary conditions and solve using solve.

```

> bcs:={bc1,bc2,bc3,bc4,bc5,bc6}:
   consts:=convert(const,set);
   Csoln:=map(simplify,solve(bcs,consts)):

```

The solutions are quite long, so we show only one example.

```

> collect(Csoln[2],[Psi0[1,1],Psi0[1,2],Psi0[2,1],
                  Psi0[2,2],p1,p2],factor);

```

$$\begin{aligned}
C2 := & \left( -\frac{1}{2} \beta^2 \omega \mu \sin(\omega \tau) p1^2 + \left( \frac{1}{2} \beta^2 \omega^2 p2 + \frac{1}{2} \mu (\delta \omega \sin(\omega \tau) + \beta p0 \mu \right. \right. \\
& \left. \left. - \beta p0 \mu \cos(\omega \tau)^2 \right) \beta \right) p1 - \frac{1}{2} \omega (\beta p0 \mu \sin(\omega \tau) + \delta \omega) \beta p2 \Big) \Psi_{01,2} \\
& + \left( \frac{1}{2} \mu \sin(\omega \tau) (-1 - \beta p0 \mu + \beta p0 \mu \cos(\omega \tau) + \omega^2) \beta p1 \right. \\
& \left. - \frac{1}{2} \omega (-1 - \beta p0 \mu + \beta p0 \mu \cos(\omega \tau) + \omega^2) \beta p2 \right) \Psi_{02,2} \\
& - \frac{1}{2} \beta p1 \mu \sin(\omega \tau) \omega + \frac{1}{2} \beta \omega^2 p2
\end{aligned}$$

The final step is to use the expressions for  $\Psi(0)$ ,  $\Phi$ , and  $\mathbf{h}$  to calculate the nonlinear terms of (8.27).

```

> fu:=collect (expand (subs (x_cm,f)), [u1,u2],
               distributed,factor);
nonlinu:=matrix ([ [0], [fu] ]):
ODE_nonlin:=multiply (Psi0,nonlinu):

```

Note that we have used the fact that the first component of the nonlinearity in our example (8.34) is 0.

Recalling our expression for the matrix  $B$ , we can see that for our example, the general equation on the centre manifold (8.27) becomes (to  $O(\|\mathbf{u}\|^3)$ )

$$\begin{aligned}
\dot{u}_1 &= \omega u_2 + f_{11}^1 u_1^2 + f_{12}^1 u_1 u_2 + f_{22}^1 u_2^2 + f_{111}^1 u_1^3 + f_{112}^1 u_1^2 u_2 \\
&\quad + f_{122}^1 u_1 u_2^2 + f_{222}^1 u_2^3, \\
\dot{u}_2 &= -\omega u_1 + f_{11}^1 u_1^2 + f_{12}^1 u_1 u_2 + f_{22}^1 u_2^2 + f_{111}^1 u_1^3 + f_{112}^1 u_1^2 u_2 \\
&\quad + f_{122}^1 u_1 u_2^2 + f_{222}^1 u_2^3.
\end{aligned} \tag{8.41}$$

The  $f_{jk}^i$  and  $f_{jkl}^i$  are functions of the parameters  $\beta, \tau, \delta, \theta, p_0, p_1, p_2$ , the Hopf frequency  $\omega$ , and the center manifold coefficients  $h_{jk}^i(0)$  and  $h_{jk}^i(-\tau)$ . As should be expected, (8.41) is an ODE at a Hopf bifurcation. The criticality of this bifurcation (and hence of the Hopf bifurcation in the original system of DDE's) may be determined by applying standard approaches. For example, one can show that the criticality of the Hopf bifurcation of (8.41) is determined by the sign of the quantity [17, p. 152]

$$\begin{aligned}
a &= \frac{1}{8} (3f_{111}^1 + f_{122}^1 + f_{112}^2 + 3f_{222}^2) \\
&\quad - \frac{1}{8\omega} (f_{12}^1 (f_{11}^1 + f_{22}^1) - f_{12}^2 (f_{11}^2 + f_{22}^2) - 2f_{11}^1 f_{11}^2 + 2f_{22}^1 f_{22}^2).
\end{aligned} \tag{8.42}$$

To evaluate this expression, we first find the coefficients of the quadratic terms.

```

> quad:=array(1..2,1..3):
> quad[1,1]:=coeff(coeff(ODE_nonlin[1,1],u1,2),u2,0);
> quad[1,2]:=coeff(coeff(ODE_nonlin[1,1],u1,1),u2,1);
> quad[1,3]:=coeff(coeff(ODE_nonlin[1,1],u1,0),u2,2);
> quad[2,1]:=coeff(coeff(ODE_nonlin[2,1],u1,2),u2,0);
> quad[2,2]:=coeff(coeff(ODE_nonlin[2,1],u1,1),u2,1);
> quad[2,3]:=coeff(coeff(ODE_nonlin[2,1],u1,0),u2,2);

```

$$\begin{aligned}
quad_{1,1} &:= 0 \\
quad_{1,2} &:= \Psi_{0,2} \omega \beta \mu p_1 (\cos(\omega \tau) - 1) \\
quad_{1,3} &:= \Psi_{0,2} \beta \omega (\omega p_2 - \mu p_1 \sin(\omega \tau)) \\
quad_{2,1} &:= 0 \\
quad_{2,2} &:= \Psi_{0,2} \omega \beta \mu p_1 (\cos(\omega \tau) - 1) \\
quad_{2,3} &:= \Psi_{0,2} \beta \omega (\omega p_2 - \mu p_1 \sin(\omega \tau))
\end{aligned}$$

The necessary cubic coefficients are found in a similar way.

```

> cub:=array(1..2,1..4):
> cub[1,1]:=coeff(coeff(ODE_nonlin[1,1],u1,3),u2,0);
> cub[1,3]:=coeff(coeff(ODE_nonlin[1,1],u1,1),u2,2):
> cub[2,2]:=coeff(coeff(ODE_nonlin[2,1],u1,2),u2,1):
> cub[2,4]:=coeff(coeff(ODE_nonlin[2,1],u1,0),u2,3);

```

$$\begin{aligned}
cub_{1,1} &:= \Psi_{0,2} \beta \mu p_1 h_{2,11}(0) (\cos(\omega \tau) - 1) \\
cub_{1,3} &:= \Psi_{0,2} \beta (\mu p_2 \cos(\omega \tau) \omega^2 - \mu p_2 \omega^2 + p_1 \mu h_{1,12}(-\tau) \omega - p_1 \mu h_{2,22}(0) \\
&\quad + 2p_2 h_{2,12}(0) \omega - p_1 \mu h_{1,12}(0) \omega - p_1 \mu \sin(\omega \tau) h_{2,12}(0) \\
&\quad + p_1 \mu \cos(\omega \tau) h_{2,22}(0))
\end{aligned}$$

$$\begin{aligned}
cub_{2,2} &:= -\Psi_{0,2} \beta (p1 \mu h_{2,12}(0) - p1 \mu h_{1,11}(-\tau) \omega p1 \mu \sin(\omega \tau) h_{2,11}(0) \\
&\quad - p1 \mu \cos(\omega \tau) h_{2,12}(0) + p1 \mu h_{1,11}(0) \omega - 2 p2 h_{2,11}(0) \omega) \\
cub_{2,4} &:= -\Psi_{0,2} \beta (\mu p2 \sin(\omega \tau) \omega^2 + p1 \mu \sin(\omega \tau) h_{2,22}(0) + p1 \mu h_{1,22}(0) \omega \\
&\quad - p1 \mu h_{1,22}(-\tau) \omega - 2 p2 h_{2,22}(0) \omega)
\end{aligned}$$

Note that only the cubic terms depend on the  $h_{jk}^i$ , as expected. The quantity  $a$  is evaluated using the formula of (8.42)

$$\begin{aligned}
a := & \text{collect}(\text{simplify}(1/8 * (3 * cub[1,1] + cub[1,3] + cub[2,2] \\
& + 3 * cub[2,4]) - 1 / (8 * \omega) * (\text{quad}[1,2] * (\text{quad}[1,1] \\
& + \text{quad}[1,3]) - \text{quad}[2,2] * (\text{quad}[2,1] + \text{quad}[2,3]) \\
& - 2 * \text{quad}[1,1] * \text{quad}[2,1] + 2 * \text{quad}[1,3] * \text{quad}[2,3]))), \\
& [\Psi_{0,1}, \Psi_{0,2}], \text{distributed}, \text{factor});
\end{aligned}$$

$$\begin{aligned}
a := & \frac{1}{64} \beta^2 \omega p1 \mu (\cos(\omega \tau) - 1) (-\omega p2 + \mu p1 \sin(\omega \tau)) \Psi_{0,1,2}^2 + \frac{1}{32} \beta^2 \omega \\
& \left( -\omega^2 p2^2 - \mu^2 p1^2 + 2 \omega p2 \mu p1 \sin(\omega \tau) + \mu^2 p1^2 (\cos(\omega \tau))^2 \right) \Psi_{0,1,2} \Psi_{0,2,2} \\
& + \frac{1}{8} \beta \left( p1 \mu \cos(\omega \tau) h_{2,22}(0) - 3 \mu p1 h_{2,11}(0) \right. \\
& + 3 \mu p1 h_{2,11}(0) \cos(\omega \tau) + 2 p2 h_{2,12}(0) \omega - p1 \mu h_{2,22}(0) \\
& + p1 \mu h_{1,12}(-\tau) \omega - p1 \mu \sin(\omega \tau) h_{2,12}(0) - \mu p2 \omega^2 - p1 \mu h_{1,12}(0) \omega \\
& + \mu p2 \cos(\omega \tau) \omega^2 \left. \right) \Psi_{0,1,2} - \frac{1}{64} \beta^2 \omega p1 \mu (\cos(\omega \tau) - 1) \left( -\omega p2 \right. \\
& + \mu p1 \sin(\omega \tau) \left. \right) \Psi_{0,2,2}^2 - \frac{1}{8} \beta \left( -2 p2 h_{2,11}(0) \omega + p1 \mu h_{2,12}(0) \right. \\
& - 6 p2 h_{2,22}(0) \omega + p1 \mu h_{1,11}(0) \omega + p1 \mu \sin(\omega \tau) h_{2,11}(0) \\
& - p1 \mu \cos(\omega \tau) h_{2,12}(0) - p1 \mu h_{1,11}(-\tau) \omega - 3 p1 \mu h_{1,22}(-\tau) \omega \\
& + 3 \mu p2 \sin(\omega \tau) \omega^2 + 3 p1 \mu h_{1,22}(0) \omega + 3 p1 \mu \sin(\omega \tau) h_{2,22}(0) \left. \right) \Psi_{0,2,2}
\end{aligned}$$

To get the final expression for  $a$  we need to substitute in the actual values for  $\Psi_0$ ,  $h_{jk}^i(0)$  and  $h_{jk}^i(\tau)$ . The expression is very large, so we do not print it out.

$$\begin{aligned}
a_{\text{final}} := & \text{subs}(\Psi_{0\_vals}, \text{simplify}(\text{subs}(\text{Csoln}, \\
& \text{simplify}(\text{subs}(\text{hsoln0}, \text{hsolnt}, a))))):
\end{aligned}$$

## 8.4 Discussion

In this chapter, we have shown how the symbolic algebra package Maple™ can be used to calculate the center manifold for a delay differential equation at a Hopf bifurcation. The commands involved are fairly simple, and thus it should be fairly easily to adapt them to other computer algebra systems.

The emphasis of this chapter was on a system at a Hopf bifurcation. The implementation for other bifurcations is similar, and just requires the modifying the following parts:

1. The calculation of the basis functions for the center eigenspace for the original and transpose systems ( $\Phi$ ,  $\Psi$ ).
2. The calculation of the quantity that determines the criticality. This depends on the normal form for the bifurcation involved.

Also, as discussed in Sect. 8.2, for some systems and some bifurcations, it may not be necessary to compute the nonlinear terms of the center manifold.

This chapter has focused on systems *at* a bifurcation. This means that our predictions of the stability of the bifurcating limit cycle will only be valid in some neighborhood of the bifurcation point. To get predictions which are valid in a larger region, one can use the approach of *parameter-dependent* center manifolds. For a general outline of this approach and some specific examples for ordinary differential equations see [17, p. 134] or [49, p. 198]. For the application of the approach to DDEs with a Hopf bifurcation see [12] or [35]. For the application of this approach to DDEs with a Bogdanov-Takens singularity see [11] or [36]. For the application of this approach to delay differential equations with a Fold-Hopf singularity see [34]. Note that the papers of Qesmi et al. [34–36] have some discussion of the implementation of their algorithms in Maple.

There are other approaches for studying the dynamics of a delay differential equation near a nonhyperbolic equilibrium point. Perturbation techniques (multiple scales, Poincaré-Lindstedt, averaging) have been used to study Hopf bifurcation [5, 8, 9, 16, 30, 37, 39, 50]. Often the computations for such methods are as extensive than for the center manifold; however, the mathematical theory is more approachable. The Liapunov-Schmidt reduction has also been implemented for delay differential equations, both numerically [1, 40–42] and using symbolic algebra [13]. However, this method only determines existence of the bifurcating solutions. Some other method must be used to determine stability.

Finally, while this chapter has focused on using center manifolds to study nonhyperbolic equilibrium points of autonomous delay differential equations (DDEs), other applications are possible. In particular, in [10, 38, 46] center manifolds are used to study DDEs with time periodic coefficients.

**Acknowledgments** Maple is a trademark of Waterloo Maple Inc. This work was supported by a grant from the Natural Sciences and Engineering Research Council of Canada. I acknowledge the contributions of my collaborators on the center manifold computations. In particular, Jacques Bélair introduced me to delay differential equations and wrote the first version of the implementation of the center manifold calculations in Maple. Emily Stone introduced me to the drilling application studied in this chapter, which motivated further development of center manifold Maple code.

## References

1. Aboud N., Sathaye A., Stech H.W. (1988) BIFDE: software for the investigation of the Hopf bifurcation problem in functional differential equations. In: Proceedings of the 27th IEEE Conference on Decision and Control. Vol. 1, pp. 821–824
2. Ait Babram M., Arino O., Hbid M.L. (1997) Approximation scheme of a system manifold for functional differential equations. *Journal of Mathematical Analysis and Applications* 213, 554–572
3. Bélair J., Campbell S.A. (1994) Stability and bifurcations of equilibria in a multiple-delayed differential equation. *SIAM Journal on Applied Mathematics* 54(5), 1402–1424
4. Campbell S.A., LeBlanc V.G. (1998) Resonant Hopf-Hopf interactions in delay differential equations. *Journal of Dynamics and Differential Equations* 10, 327–346
5. Campbell S.A., Ncube I., Wu J. (2006) Multistability and stable asynchronous periodic oscillations in a multiple-delayed neural system. *Physica D* 214(2), 101–119
6. Campbell S.A., Yuan Y. (2008) Zero singularities of codimension two and three in delay differential equations. *Nonlinearity* 21, 2671–2691
7. Campbell S.A., Yuan Y., Bungay S.D. (2005) Equivariant Hopf bifurcation in a ring of identical cells with delayed coupling. *Nonlinearity* 18, 2827–2846
8. Chow S.-N., Mallet-Paret J. (1977) Integral averaging and Hopf bifurcation. *Journal of Differential Equations* 26, 112–159
9. Das S.L., Chatterjee A. (2002) Multiple scales without center manifold reductions for delay differential equations near Hopf bifurcations. *Nonlinear Dynamics* 30(4), 323–335
10. Deshmukh V., Butcher E.A., Bueler E. (2008) Dimensional reduction of nonlinear delay differential equations with periodic coefficients using Chebyshev spectral collocation. *Nonlinear Dynamics* 52, 137–149
11. Faria T., Magalhães L. (1995a) Normal forms for retarded functional differential equations with parameters and applications to Bogdanov-Takens singularity. *Journal of Differential Equations* 122, 201–224
12. Faria T., Magalhães L. (1995b) Normal forms for retarded functional differential equations with parameters and applications to Hopf bifurcation. *Journal of Differential Equations* 122, 181–200
13. Franke J.M., Stech H.W. (1991) Extensions of an algorithm for the analysis of nongeneric Hopf bifurcations, with applications to delay-difference equations. In: Busenberg S, Martelli M (eds), *Delay Differential Equations and Dynamical Systems*. Vol. 1475 of Springer Lecture Notes in Mathematics. Springer-Verlag, Berlin, pp. 161–175
14. Gilsinn D.E. (2002) Estimating critical Hopf bifurcation parameters for a second order delay differential equation with application to machine tool chatter. *Nonlinear Dynamics* 30, 103–154
15. Gilsinn D.E. (2008) Bifurcations, center manifolds, and periodic solutions. In: Balachandran B, Gilsinn DE, Kalmár-Nagy T (eds), *Delay Differential Equations: Recent Advances and New Directions*. Springer Verlag, New York, pp. 157–204
16. Gopalsamy K., Leung I. (1996) Delay induced periodicity in a neural netlet of excitation and inhibition. *Physica D* 89, 395–426
17. Guckenheimer J., Holmes P.J. (1983) *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*. Springer-Verlag, New York
18. Guo S. (2005) Spatio-temporal patterns of nonlinear oscillations in an excitatory ring network with delay. *Nonlinearity* 18, 2391–2407
19. Guo S., Huang L. (2003) Hopf bifurcating periodic orbits in a ring of neurons with delays. *Physica D* 183, 19–44
20. Guo S., Huang L., Wang L. (2004) Linear stability and Hopf bifurcation in a two neuron network with three delays. *International Journal of Bifurcation and Chaos* 14, 2799–2810
21. Hale J.K. (1985) Flows on center manifolds for scalar functional differential equations. *Proceedings of the Royal Society of Edinburgh* 101A, 193–201

22. Hale J.K., Verduyn Lunel S.M. (1993) Introduction to Functional Differential Equations. Springer Verlag, New York
23. Jiang M., Shen Y., Jian J., Liao X. (2006) Stability, bifurcation and a new chaos in the logistic differential equation with delay. *Physics Letters A* 350(3–4), 221–227
24. Kalmár-Nagy T., Pratt J.R., Davies M.A., Kennedy M.D. (1999) Experimental and analytical investigation of the subcritical instability in turning. In: Proceedings of the 1999 ASME Design Engineering Technical Conferences, 17th ASME Biennial Conference on Mechanical Vibration and Noise. DECT99/VIB-8060
25. Kalmár-Nagy T., Stépán G., Moon F.C. (2001) Subcritical Hopf bifurcation in the delay equation model for machine tool vibrations. *Nonlinear Dynamics* 26, 121–142
26. Krawcewicz W., Wu J. (1999) Theory and applications of Hopf bifurcations in symmetric functional-differential equations. *Nonlinear Analysis* 35 (7, Series A: Theory Methods), 845–870
27. Landry M., Campbell S.A., Morris K.A., Aguilar C. (2005) Dynamics of an inverted pendulum with delayed feedback control. *SIAM Journal on Applied Dynamical Systems* 4 (2), 333–351
28. Liu Z., Yuan R. (2005) Stability and bifurcation in a harmonic oscillator with delays. *Chaos, Solitons and Fractals* 23, 551–562
29. Maple 9.5 Getting Started Guide (2004) Maplesoft, a division of Waterloo Maple Inc., Toronto, Canada
30. Nayfeh A.H. (2008) Order reduction of retarded nonlinear systems – the method of multiple scales versus center-manifold reduction. *Nonlinear Dynamics* 51, 483–500
31. Orosz G., Stépán G. (2004) Hopf bifurcation calculations in delayed systems with translational symmetry. *Journal of Nonlinear Science* 14(6), 505–528
32. Orosz G., Stépán G. (2006) Subcritical Hopf bifurcations in a car-following model with reaction-time delay. *Proceedings of the Royal Society of London, series A* 462(2073), 2643–2670
33. Perko L. (1996) Differential Equations and Dynamical Systems. Springer-Verlag, New York
34. Qesmi R., Ait Babram M., Hbid M.L. (2006a) Center manifolds and normal forms for a class of retarded functional differential equations with parameter associated with Fold-Hopf singularity. *Applied Mathematics and Computation* 181(1), 220–246
35. Qesmi R., Ait Babram M., Hbid M.L. (2006b) Computation of terms of center manifolds and normal elements of bifurcations for a class of functional differential equations associated with Hopf singularity. *Applied Mathematics and Computation* 175(2), 932–968
36. Qesmi R., Ait Babram M., Hbid M.L. (2007) Symbolic computation for center manifolds and normal forms of Bogdanov bifurcation in retarded functional differential equations. *Nonlinear Analysis* 66, 2833–2851
37. Rand R., Verdugo A. (2007) Hopf bifurcation formula for first order differential-delay equations. *Communications in Nonlinear Science and Numerical Simulation* 12(6), 859–864
38. Sri Namachchivaya N., van Roessel H.J. (2003) A centre-manifold analysis of variable speed machining. *Dynamical Systems* 18(3), 245–270
39. Stech H.W. (1979) The Hopf bifurcation: a stability result and application. *Journal of Mathematical Analysis and Applications* 71, 525–546
40. Stech H.W. (1985a) Hopf bifurcation analysis in a class of scalar functional differential equations. In: Lighthorne J, Rankin S (eds), *Physical mathematics and nonlinear partial differential equations*. Marcel Dekker, New York, pp. 175–186
41. Stech H.W. (1985b) Hopf bifurcation calculations for functional differential equations. *Journal of Mathematical Analysis and Applications* 109, 472–491
42. Stech H.W. (1985c) Nongeneric Hopf bifurcations in functional differential equations. *SIAM Journal on Mathematical Analysis* 16, 1134–1151
43. Stépán G., Haller G. (1995) Quasiperiodic oscillations in robot dynamics. *Nonlinear Dynamics* 8, 513–528
44. Stone E., Askari A. (2002) Nonlinear models of chatter in drilling processes. *Dynamical Systems* 17(1), 65–85

45. Stone E., Campbell S.A. (2004) Stability and bifurcation analysis of a nonlinear DDE model for drilling. *Journal of Nonlinear Science* 14(1), 27–57
46. Szalai R., Stépán G. (2005) Period doubling bifurcation and center manifold reduction in a time-periodic and time-delayed model of machining, preprint
47. Verdugo A., Rand R. (2008) Center manifold analysis of a DDE model of gene expression. *Communications in Nonlinear Science and Numerical Simulation* 13(6), 1112–1120
48. Wei J.J., Yuan Y. (2005) Synchronized Hopf bifurcation analysis in a neural network model with delays. *Journal of Mathematical Analysis and Applications* 312(1), 205–229
49. Wiggins S. (1990) *Introduction to Applied Nonlinear Dynamic Systems and Chaos*. Springer Verlag, New York
50. Wirkus S., Rand R. (2004) The dynamics of two coupled van der Pol oscillators with delay coupling. *Nonlinear Dynamics* 30(3), 205–221
51. Wischert W., Wunderlin A., Pelster A., Olivier M., Grosblambert J (1994) Delay-induced instabilities in nonlinear feedback systems. *Physical Review E* 49(1), 203–219
52. Wu J. (1998) Symmetric functional-differential equations and neural networks with memory. *Transactions of the American Mathematical Society* 350(12), 4799–4838
53. Wu J., Faria T., Huang Y.S. (1999) Synchronization and stable phase-locking in a network of neurons with memory. *Mathematical and Computer Modelling* 30(1–2), 117–138
54. Yuan Y., Campbell S.A. (2004) Stability and synchronization of a ring of identical cells with delayed coupling. *Journal of Dynamics and Differential Equations* 16(1), 709–744
55. Yuan Y., Wei J.J. (2005) Multiple bifurcation analysis in a neural network model with delays. *International Journal of Bifurcation and Chaos* 16(10), 2903–2913
56. Yuan Y., Yu P., Librescu L., Marzocca P. (2004) Aeroelasticity of time-delayed feedback control of two-dimensional supersonic lifting surfaces. *Journal of Guidance, Control, and Dynamics* 27(5), 795–803

# Chapter 9

## Numerical Solution of Delay Differential Equations

Larry F. Shampine and Sylvester Thompson

**Abstract** After some introductory examples, in this chapter, some of the ways in which delay differential equations (DDEs) differ from ordinary differential equations (ODEs) are considered. Then, numerical methods for DDEs are discussed, and in particular, how the Runge–Kutta methods that are so popular for ODEs can be extended to DDEs. The treatment of these topics is complete, but it is necessarily brief, so it would be helpful to have some background in the theory of ODEs and their numerical solution. The chapter goes on to consider software issues special to the numerical solution of DDEs and concludes with some substantial numerical examples. Both topics are discussed in concrete terms using the programming languages MATLAB and Fortran 90/95, so a familiarity with one or both languages would be helpful.

**Keywords:** DDEs · Propagated discontinuities · Vanishing delays · Numerical methods · Runge–Kutta · Continuous extension · Event location · MATLAB · Fortran 90/95

### 9.1 Introduction

Ordinary differential equations (ODEs) have been used to model physical phenomena since the concept of differentiation was first developed, and nowadays complicated ODE models can be solved numerically with a high degree of confidence. It was recognized early that phenomena may have a delayed effect in a differential equation, leading to what is called a *delay differential equation* (DDE). For instance, fishermen along the west coast of South America have long observed a sporadic and abrupt warming of the cold waters that support the food chain. The recent investigation [10] of this El-Niño/Southern Oscillation (ENSO) phenomenon discusses the history of models starting in the 1980s that account for delayed feedbacks. An early model of this kind,

$$T'(t) = T(t) - \alpha T(t - \tau) \quad (9.1)$$

for constant  $\alpha > 0$ , is simple enough to study analytically. It is the term involving a constant *lag* or *delay*  $\tau > 0$  in the independent variable that makes this a DDE. An obvious distinction between this DDE and an ODE is that specifying the initial value  $T(0)$  is not enough to determine the solution for  $t \geq 0$ ; it is necessary to specify the *history*  $T(t)$  for  $-\tau \leq t \leq 0$  for the differential equation even to be defined for  $0 \leq t \leq \tau$ . The paper [10] goes on to develop and study a more elaborate nonlinear model with periodic forcing and a number of physical parameters of the form

$$h'(t) = -a \tanh[\kappa h(t - \tau)] + b \cos(2\pi \omega t). \quad (9.2)$$

These models exemplify DDEs with constant delays. The first mathematical software for solving DDEs is `dmrde` [17], which did not appear until 1975. Today a good many programs that can solve reliably first-order systems of DDEs with constant delays are available, though the paper [10] makes clear that even for this relatively simple class of DDEs, there can be serious computational difficulties. This is not just a matter of developing software, rather that DDEs have more complex behavior than ODEs. Some numerical results for (9.2) are presented in Sect. 9.4.1.

Some models have delays  $\tau_j(t)$  that depend on time. Provided that the delays are bounded away from zero, the models behave similarly to those with constant delays and they can be solved with some confidence. However, if a delay goes to zero, the differential equation is said to be *singular* at that time. Such singular problems with *vanishing delays* present special difficulties in both theory and practice. As a concrete example of a problem with two time-dependent delays, we mention one that arises from delayed cellular neural networks [31]. The fact that the delays are sometimes very small and even vanish periodically during the integration makes this a relatively difficult problem.

$$\begin{aligned} y_1'(t) &= -6y_1(t) + \sin(2t)f(y_1(t)) + \cos(3t)f(y_2(t)) \\ &\quad + \sin(3t)f\left(y_1\left(t - \frac{1 + \cos(t)}{2}\right)\right) + \sin(t)f\left(y_2\left(t - \frac{1 + \sin(t)}{2}\right)\right) \\ &\quad + 4\sin(t) \\ y_2'(t) &= -7y_2(t) + \frac{\cos(t)}{3}f(y_1(t)) + \frac{\cos(2t)}{2}f(y_2(t)) \\ &\quad + \cos(t)f\left(y_1\left(t - \frac{1 + \cos(t)}{2}\right)\right) + \cos(2t)f\left(y_2\left(t - \frac{1 + \sin(t)}{2}\right)\right) \\ &\quad + 2\cos(t). \end{aligned}$$

Here  $f(x) = (|x+1| - |x-1|)/2$ . The problem is defined by this differential equation and the history  $y_1(t) = -0.5$  and  $y_2(t) = 0.5$  for  $t \leq 0$ . A complication with this particular example is that there are time-dependent impulses, but we defer discussion of that issue to §9.4.3 where we solve it numerically as in [6]. Some models have delays that depend on the solution itself as well as time,  $\tau(t, y)$ . Not surprisingly, it is more difficult to solve such problems because only an approximation to the solution is available for defining the delays.

Now that we have seen some concrete examples of DDEs, let us state more formally the equations that we discuss in this ninth chapter. In a first-order system of ODEs

$$y'(t) = f(t, y(t)), \quad (9.3)$$

the derivative of the solution depends on the solution at the present time  $t$ . In a first-order system of DDEs, the derivative also depends on the solution at earlier times. As seen in the extensive bibliography [2], such problems arise in a wide variety of fields. In this chapter, we consider DDEs of the form

$$y'(t) = f(t, y(t), y(t - \tau_1), y(t - \tau_2), \dots, y(t - \tau_k)). \quad (9.4)$$

Commonly the delays  $\tau_j$  here are positive constants. There is, however, considerable and growing interest in systems with time-dependent delays  $\tau_j(t)$  and systems with state-dependent delays  $\tau_j(t, y(t))$ . Generally, we suppose that the problem is non-singular in the sense that the delays are bounded below by a positive constant,  $\tau_j \geq \tau > 0$ . We shall see that it is possible to adapt methods for the numerical solution of initial value problems (IVPs) for ODEs to the solution of initial value problems for DDEs. This is not straightforward because DDEs and ODEs differ in important ways. Equations of the form (9.4), even with time- and state-dependent delays, do not include all the problems that arise in practice. Notably absent are equations that involve a derivative with delayed argument like  $y'(t - \tau_m)$  on the right-hand side. Equations with such terms are said to be of *neutral* type. Though we comment on neutral equations in passing, we study in this ninth chapter just DDEs of the form (9.4). We do this because neutral DDEs can have quite different behavior that is numerically challenging. Although we cite programs that have reasonable prospects for solving a neutral DDE, the numerical solution of neutral DDEs is still a research area.

## 9.2 DDEs are not ODEs

In this section, we consider some of the most important differences between DDEs and ODEs. A fundamental technique for solving a system of DDEs is to reduce it to a sequence of ODEs. This technique and other important methods for solving DDEs are illustrated.

The simple ENSO model (9.1) is a constant coefficient, homogeneous differential equation. If it were an ODE, we might solve it by looking for solutions of the form  $T(t) = e^{\lambda t}$ . Substituting this form into the ODE leads to an algebraic equation, the characteristic equation, for values  $\lambda$  that provide a solution. For a first-order equation, there is only one such value. The same approach can be applied to DDEs. Here it leads first to

$$\lambda e^{\lambda t} = -\alpha e^{\lambda(t-\tau)} + e^{\lambda t}$$

and then to the characteristic equation

$$\lambda = -\alpha e^{-\lambda \tau} + 1$$

In contrast to the situation with a first-order ODE, this algebraic equation has infinitely many roots  $\lambda$ . Asymptotic expressions for the roots of large modulus are derived in [9]. They show that the equation can have solutions that oscillate rapidly. To make the point more concretely, we consider an example from [9] for which it is easy to determine the roots analytically, even with a parameter  $a$ , namely the DDE of neutral type

$$y'(t) = y'(t - \tau) + a(y(t) - y(t - \tau)). \quad (9.5)$$

Substituting  $y(t) = e^{\lambda t}$  into this equation leads to

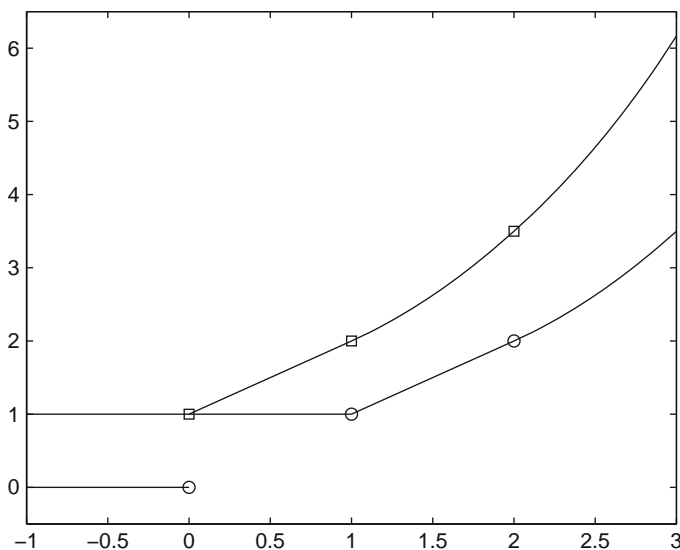
$$(\lambda - a)(1 - e^{-\lambda \tau}) = 0.$$

The two real roots 0 and  $a$  are obvious. They correspond to a constant solution and an exponential solution  $y(t) = e^{at}$ , respectively. These solutions are not surprising because they are like what we might find with an ODE. However, any  $\lambda \tau$  leading to a root of unity also provides a solution. Written in terms of real functions, we find then that there are solutions  $\cos(2\pi n t / \tau)$  and  $\sin(2\pi n t / \tau)$  for *any* integer  $n$ . This is surprising because it is so different from the behavior possible with an ODE.

These observations about homogeneous DDEs with constant coefficients and constant delays show that they can have solutions that behave quite differently from ODEs. The *method of steps* is a basic technique for studying DDEs that reduces them to a sequence of ODEs. To show how it goes and to illustrate other differences between ODEs and DDEs, we solve

$$y'(t) = y(t - 1) \quad (9.6)$$

with history  $S(t) = 1$  for  $t \leq 0$ . On the interval  $0 \leq t \leq 1$ , the function  $y(t - 1)$  in (9.6) has the known value  $S(t - 1) = 1$  because  $t - 1 \leq 0$ . Accordingly, the DDE on this interval reduces to the ODE  $y'(t) = 1$  with initial value  $y(0) = S(0) = 1$ . We solve this IVP to obtain  $y(t) = t + 1$  for  $0 \leq t \leq 1$ . Notice that the solution of the DDE exhibits a typical discontinuity in its first derivative at  $t = 0$  because  $S'(0) = 0 = y'(0-)$  and  $y'(0+) = 1$ . Now that we know the solution for  $t \leq 1$ , we can reduce the DDE on the interval  $1 \leq t \leq 2$  to an ODE  $y' = (t - 1) + 1 = t$  with initial value  $y(1) = 2$  and solve this IVP to find that  $y(t) = 0.5t^2 + 1.5$  on this interval. The first derivative is continuous at  $t = 1$ , but there is a discontinuity in the second derivative. It is straightforward to see that the solution of the DDE on the interval  $[k, k + 1]$  is a polynomial of degree  $k + 1$  and it has a discontinuity of order  $k + 1$  at time  $t = k$ . By a discontinuity of order  $k + 1$  at a time  $t = t^*$ , we mean that  $y^{(k+1)}$  has a jump there. Figure 9.1 illustrates these observations. The upper (continuous) curve with square markers at integers is the solution  $y(t)$  and the lower curve with circle markers is the derivative  $y'(t)$ . The jump from a constant value of 0 in the derivative for  $t < 0$  to a value of 1 at  $t = 0$  leads to a sharp change in the solution there. The discontinuity propagates to  $t = 1$  where the derivative has a sharp change and the solution has a less obvious change in its concavity. The jump in the third derivative at  $t = 2$  is not noticeable in the plot of  $y(t)$ .



**Fig. 9.1** Solution smoothing

In principle we can proceed in a similar way with the general equation (9.4) for delays that are bounded away from zero,  $\tau_j \geq \tau > 0$ . With the history function  $S(t)$  defined for  $t \leq t_0$ , the DDEs reduce to ODEs on the interval  $[t_0, t_0 + \tau]$  because for each  $j$ , the argument  $t - \tau_j \leq t - \tau \leq t_0$  and the  $y(t - \tau_j)$  have the known values  $S(t - \tau_j)$ . Thus, we have an IVP for a system of ODEs with initial value  $y(t_0) = S(t_0)$ . We solve this problem on  $[t_0, t_0 + \tau]$  and extend the definition of  $S(t)$  to this interval by taking it to be the solution of this IVP. Now that we know the solution for  $t \leq t_0 + \tau$ , we can move on to the interval  $[t_0 + \tau, t_0 + 2\tau]$ , and so forth. In this way, we can see that the DDEs have a unique solution on the whole interval of interest by solving a sequence of IVPs for ODEs. As with the simple example, there is generally a discontinuity in the first derivative at the initial point. If a solution of (9.4) has a discontinuity at the time  $t^*$  of order  $k$ , then as the variable  $t$  moves through  $t^* + \tau_j$ , there is a discontinuity in  $y^{(k+1)}$  because of the term  $y(t - \tau_j)$  in the DDEs. With multiple delays, a discontinuity at the time  $t^*$  is propagated to the times

$$t^* + \tau_1, t^* + \tau_2, \dots, t^* + \tau_k$$

and each of these discontinuities is in turn propagated. If there is a discontinuity at the time  $t^*$  of order  $k$ , the discontinuity at each of the times  $t^* + \tau_j$  is of order at least  $k + 1$ , and so on. This is a fundamental distinction between DDEs and ODEs: There is normally a discontinuity in the first derivative at the initial point and it is propagated throughout the interval of interest. Fortunately, for problems of the form (9.4) the solution becomes smoother as the integration proceeds. That is not the case with neutral DDEs, which is one reason that they are so much more difficult.

Neves and Feldstein [18] characterize the propagation of derivative discontinuities. The times at which discontinuities occur form a *discontinuity tree*. If there is a derivative discontinuity at  $T$ , the equation (9.4) shows that there will generally be a discontinuity in the derivative of one higher order if for some  $j$ , the argument  $t - \tau_j(t, y(t)) = T$  because the term  $y(t - \tau_j(t, y(t)))$  has a derivative discontinuity at  $T$ . Accordingly, the times at which discontinuities occur are zeros of functions

$$t - \tau_j(t, y(t)) - T = 0. \quad (9.7)$$

It is required that the zeros have odd multiplicity so that the delayed argument actually crosses the previous jump point and in practice, it is always assumed that the multiplicity is one. Although the statement in [18] is rather involved, the essence is that if the delays are bounded away from zero and delayed derivatives are not present, a derivative discontinuity is propagated to a discontinuity in (at least) the next higher derivative. In contrast, smoothing does not necessarily occur for neutral problems nor for problems with vanishing delays. For constant and time-dependent delays, the discontinuity tree can be constructed in advance. If the delay depends on the state  $y$ , the points in the tree are located by solving the algebraic equations (9.7). A very important practical matter is that the solvers have to track only discontinuities with order lower than the order of the integration method because the behavior of the method is not affected directly by discontinuities in derivatives of higher order.

Multiple delays cause special difficulties. Suppose, for example, that one delay is 1 and another is 0.001. A discontinuity in the first derivative at the initial point  $t = 0$  propagates to 0.001, 0.002, 0.003, ... because of the second delay. These “short” delays are troublesome, but the orders of the discontinuities increase and soon they do not trouble numerical methods. However, the other delay propagates the initial discontinuity to  $t = 1$  and the discontinuity there is then propagated to 1.001, 1.002, 1.003, ... because of the second delay. That is, the effects of the short delay die out, but they recur because of the longer delay. Another difficulty is that discontinuities can cluster. Suppose that one delay is 1 and another is  $1/3$ . The second delay causes an initial discontinuity at  $t = 0$  to propagate to  $1/3, 2/3, 3/3, \dots$  and the first delay causes it to propagate to  $t = 1, \dots$ . In principle the discontinuity at  $3/3$  occurs at the same time as the one at 1, but  $1/3$  is not represented exactly in finite precision arithmetic, so it is found that in practice there are two discontinuities that are extremely close together. This simple example is an extreme case, but it shows how innocuous delays can lead to clustering of discontinuities, clearly a difficult situation for numerical methods.

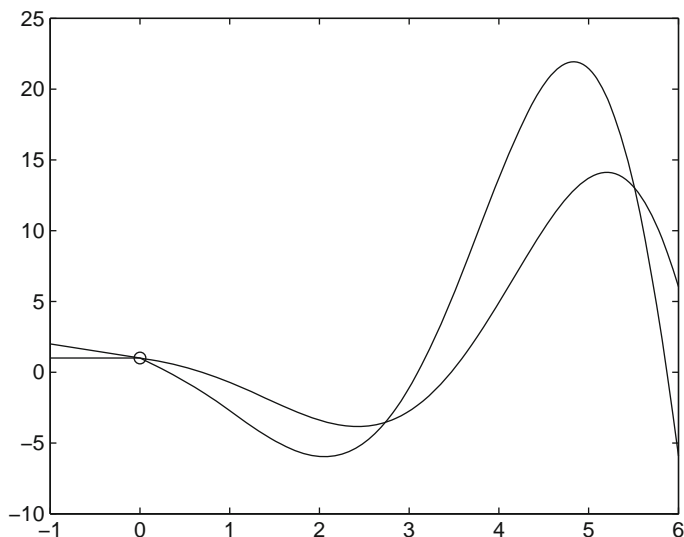
There is no best way to solve DDEs and as a result, a variety of methods that have been used for ODEs have been modified for DDEs and implemented in modern software. It is possible, though awkward in some respects, to adapt linear multistep methods for DDEs. This approach is used in the `snddelm` solver [15], which is based on modified Adams methods [24]. There are a few solvers based on implicit Runge–Kutta methods. Several are described in [12, 14, 25], including two codes, `radar5` and `ddesd`, that are based on Radau IIA collocation methods. By far the most popular approach to nonstiff problems is to use explicit Runge–Kutta methods [3, 13]. Widely used solvers include `archi` [21, 22], `dde23` [4, 29], `ddverk` [7, 8],

and `dde_solver` [5, 30]. Because the approach is so popular, it is the one that we discuss in this chapter. Despite sharing a common approach, the codes cited deal with important issues in quite different ways.

Before taking up numerical issues and how they are resolved, we illustrate the use of numerical methods by solving the model (9.1) over  $[0, 6]$  for  $\alpha = 2$  and  $\tau = 1$  with two history functions,  $T(t) = 1 - t$  and  $T(t) = 1$ . By exploiting features of the language, the MATLAB [16] solvers `dde23` and `ddesd` and the Fortran 90/95 solver `dde_solver` make it nearly as easy to solve DDEs as ODEs. Because of this and because we are very familiar with them, we use these solvers for all our numerical examples. A program that solves the DDE (9.1) with both histories and plots the results is

```
function Ex1
lags = 1; tspan = [0 6];
sol1 = dde23(@dde, lags, @history, tspan);
sol2 = dde23(@dde, lags, 1, tspan);
tplot = linspace(0, 6, 100);
T1 = deval(sol1, tplot);
T2 = deval(sol2, tplot);
% Add linear histories to the plots:
tplot = [-1 tplot]; T1 = [1 T1]; T2 = [2 T2];
plot(tplot, T1, tplot, T2, 0, 1, 'o')
%--Subfunctions-----
function dydt = dde(t, T, Z)
dydt = T - 2*Z;
function s = history(t)
s = 1 - t;
```

The output of this program is displayed as Fig. 9.2. Although MATLAB displays the output in color, all the figures of this chapter are monochrome. Except for the additional information needed to define a delay differential equation, solving a DDE with constant delays using `dde23` is nearly the same as solving an ODE with `ode23`. The first argument tells the solver of the function for evaluating the DDEs (9.4). The `lags` argument is a vector of the lags  $\tau_1, \dots, \tau_k$ . There is only one lag in the example DDE, so the argument is here a scalar. The argument `tspan` specifies the interval  $[t_0, t_f]$  of interest. Unlike the ODE solvers of MATLAB, the DDE solvers require that  $t_0 < t_f$ . The third argument is a function for evaluating the history, i.e., the solution for  $t \leq t_0$ . A constant history function is so common that the solver allows users to supply a constant vector instead of a function and that was done in computing the second solution. The first two arguments of the function for evaluating the DDEs is the same as for a system of ODEs, namely the independent variable  $t$  and a column vector of dependent variables approximating  $y(t)$ . Here the latter is called `T` and is a scalar. A challenging task for the user interface is to accommodate multiple delays. This is done with the third argument `Z`, which is an array of  $k$  columns. The first column approximates  $y(t - \tau_1)$  with  $\tau_1$  defined as the first delay in `lag`. The second column corresponds to the second delay, and so forth. Here there is only one



**Fig. 9.2** The simple ENSO model (9.1) for two histories

delay and only one dependent variable, so  $Z$  is a scalar. A notable difference between `dde23` and `ode23` is the output, here called `sol`. The ODE solvers of MATLAB optionally return solutions as a complex data structure called a *structure*, but that is the *only* form of output from the DDE solvers. The solution structure `sol1` returned by the first integration contains the mesh that was selected as the field `sol1.x` and the solution at these points as the field `sol1.y`. With this information the first solution can be plotted by `plot(sol1.x, sol1.y)`. Often this is satisfactory, but this particular problem is so easy that values of the solution at mesh points alone does not provide a smooth graph. Approximations to the solution can be obtained anywhere in the interval of interest by means of the auxiliary function `deval`. In the example program, the two solutions are approximated at 100 equally spaced points for plotting. A marker is plotted at  $(0, T(0))$  to distinguish the two histories from the corresponding computed solutions of (9.1). The discontinuity in the first derivative at the initial point is clear for the history  $T(t) = 1$ , but the discontinuity in the second derivative at  $t = 1$  is scarcely visible.

### 9.3 Numerical Methods and Software Issues

The method of steps and modifications of methods for ODEs can be used to solve DDEs. Because they are so popular, we study in this chapter only explicit Runge–Kutta methods. In addition to discussing basic algorithms, we take up important issues that arise in designing software for DDEs.

### 9.3.1 Explicit Runge–Kutta Methods

Given  $y_n \approx y(t_n)$ , an explicit Runge–Kutta method for a first-order system of ODEs (9.3) takes a step of size  $h_n$  to form  $y_{n+1} \approx y(t_{n+1})$  at  $t_{n+1} = t_n + h_n$  as follows. It first evaluates the ODEs at the beginning of the step,

$$y_{n,1} = y_n, \quad f_{n,1} = f(t_n, y_{n,1}),$$

and then for  $j = 2, 3, \dots, s$  forms

$$y_{n,j} = y_n + h_n \sum_{k=1}^{j-1} \beta_{j,k} f_{n,k}, \quad f_{n,j} = f(t_n + \alpha_j h_n, y_{n,j})$$

It finishes with

$$y_{n+1} = y_n + h_n \sum_{k=1}^s \gamma_k f_{n,k}.$$

The constants  $s$ ,  $\beta_{j,k}$ ,  $\alpha_j$ , and  $\gamma_k$  are chosen to make  $y_{n+1}$  an accurate approximation to  $y(t_{n+1})$ . The numerical integration of a system of ODEs starts with given initial values  $y_0 = y(t_0)$  and then forms a sequence of approximations  $y_0, y_1, y_2, \dots$  to the solution at times  $t_0, t_1, t_2, \dots$  that span the interval of interest,  $[t_0, t_f]$ . For some purposes, it is important to approximate the solution at times  $t$  that are not in the mesh when solving a system of ODEs, but this is crucial to the solution of DDEs. One of the most significant developments in the theory and practice of Runge–Kutta methods is a way to approximate the solution accurately and inexpensively throughout the span of a step, i.e., anywhere in  $[t_n, t_{n+1}]$ . This is done with a companion formula called a *continuous extension*.

A natural and effective way to solve DDEs can be based on an explicit Runge–Kutta method and the method of steps. The first difficulty is that in taking a step, the function  $f$  must be evaluated at times  $t_n + \alpha_j h_n$ , which for the delayed arguments means that we need values of the solution at times  $(t_n + \alpha_j h_n) - \tau_m$ , which may be prior to  $t_n$ . Generally, these times do not coincide with mesh points, so we obtain the solution values from an interpolant. For this purpose, linear multistep methods are attractive because the popular methods all have natural interpolants that provide accurate approximate solutions between mesh points. For other kinds of methods, it is natural to use Hermite interpolation for this purpose. However, though formally correct as the step size goes to zero, the approach does not work well with Runge–Kutta methods. That is because these methods go to considerable expense to form an accurate approximation at the end of the step and as a corollary, an efficient step size is often too large for accurate results from direct interpolation of solution and first derivative at several previous steps. Nowadays, codes based on Runge–Kutta formulas use a continuous extension of the basic formula that supplements the function evaluations formed in taking the step to obtain accurate approximations throughout  $[t_n, t_{n+1}]$ . This approach uses only data from the current interval. With any of these approaches, we obtain a polynomial interpolant that approximates the solution over the span of a step and in aggregate, the interpolants form a piecewise-polynomial

function that approximates  $y(t)$  from  $t_0$  to the current  $t_n$ . Care must be taken to ensure that the accuracy of the interpolant reflects that of the basic formula and that various polynomials connect at mesh points in a sufficiently smooth way. Details can be found in [3] and [13]. It is important to appreciate that interpolation is used for other purposes, too. We have already seen it used to get the smooth graph of Fig. 9.2. It is all but essential when solving (9.7). Capable solvers also use interpolation for *event location*, a matter that we discuss in Sect. 9.3.3.

Short delays arise naturally and as we saw by example in §9.1, it may even happen that a delay vanishes. If no delayed argument occurs prior to the initial point, the DDE has no history, so it is called an *initial value* DDE. A simple example is  $y'(t) = y(t^2)$  for  $t \geq 0$ . A delay that vanishes can lead to quite different behavior – the solution may not extend beyond the singular point or it may extend, but not be unique. Even when the delays are all constant, “short” delays pose an important practical difficulty: Discontinuities smooth out as the integration progresses, so we may be able to use a step size much longer than a delay. In this situation, an explicit Runge–Kutta formula needs values of the solution at “delayed” arguments that are in the span of the current step. That is, we need an approximate solution at points in  $[t_n, t_{n+1}]$  before  $y_{n+1}$  has been computed. Early codes simply restrict the step size so as to avoid this difficulty. However, the most capable solvers predict a solution throughout  $[t_n, t_{n+1}]$  using the continuous extension from the preceding step. They compute a tentative  $y_{n+1}$  using predicted values for the solution at delayed arguments and then repeat using a continuous extension for the current step until the values for  $y_{n+1}$  converge. This is a rather interesting difference between ODEs and DDEs: If the step size is bigger than a delay, a Runge–Kutta method that is explicit for ODEs is generally an implicit formula for  $y_{n+1}$  for DDEs. The iterative scheme for taking a step resembles a predictor–corrector iteration for evaluating an implicit linear multistep method. More details are available in [1, 29].

The order of accuracy of a Runge–Kutta formula depends on the smoothness of the solution in the span of the step. This is not usually a problem with ODEs, but we have seen that discontinuities are to be expected when solving DDEs. To maintain the order of the formula, we have to locate and step to discontinuities. The popular codes handle propagated discontinuities in very different ways. The solver `dde23` allows only constant delays, so before starting the integration, it determines all the discontinuities in the interval of interest and arranges for these points to be included in the mesh  $t_0, t_1, \dots$ . Something similar can be done if the delays are time-dependent, but this is awkward, especially if there are many delays. The approach is not applicable to problems with state-dependent delays. For problems of this generality, discontinuities are located by solving (9.7) as the integration proceeds. If the function (9.7) changes sign between  $t_n$  and  $t_{n+1}$ , the algebraic equation is solved for the location of the discontinuity with the polynomial interpolant for this step,  $P(t)$ , replacing  $y(t)$  in (9.7). After locating the first time  $t^*$  at which  $\alpha(t^*, P(t^*)) - T = 0$ , the step to  $t_{n+1}$  is rejected and a shorter step is taken from  $t_n$  to a new  $t_{n+1} = t^*$ . This is what `dde_solver` does for general delays, but it uses the more efficient approach of building the discontinuity tree in advance for problems with constant delays. `archi` also solves (9.7), but tracking discontinuities is an option in this solver. `ddverk`

does not track discontinuities explicitly, rather it uses the *defect* or *residual* of the solution to detect discontinuities. The residual  $r(t)$  of an approximate solution  $S(t)$  is the amount by which it fails to satisfy the differential equation:

$$S'(t) = f(t, S(t), S(t - \tau_1), S(t - \tau_2), \dots, S(t - \tau_k)) + r(t).$$

On discovering a discontinuity, `ddverk` steps across with special interpolants. `ddesd` does not track propagated discontinuities. Instead it controls the residual, which is less sensitive to the effects of propagated discontinuities. `radar5` treats the step size as a parameter to locate discontinuities detected by error test failures; details are found in [11].

When solving a system of DDEs, it is generally necessary to supply an initial history function to provide solution values for  $t \leq t_0$ . Of course the history function must provide values for  $t$  as far back from  $t_0$  as the maximum delay. This is straightforward, but some DDEs have discontinuities at times prior to the initial point or even at the initial point. A few solvers, including `dde23`, `ddesd`, and `dde_solver`, provide for this, but most codes do not because it complicates both the user interface and the program.

### 9.3.2 Error Estimation and Control

Several quite distinct approaches to the vital issue of error estimation and control are seen in popular solvers. In fact, this issue most closely delineates the differences between DDE solvers. Popular codes like `archi`, `dde23`, `dde_solver`, and `ddverk` use pairs of formulas for this purpose. The basic idea is to take each step with two formulas and estimate the error in the lower order result by comparison. The cost is kept down by embedding one formula in the other, meaning that one formula uses only  $f_{n,k}$  that were formed in evaluating the other formula or at most a few extra function evaluations. The error of the lower order formula is estimated and controlled, but it is believed that the higher order formula is more accurate, so most codes advance the integration with the higher order result. This is called *local extrapolation*. In some ways an embedded pair of Runge–Kutta methods is rather like a predictor–corrector pair of linear multistep methods.

If the pair is carefully matched, an efficient and reliable estimate of the error is obtained when solving a problem with a smooth solution. Indeed, most ODE codes rely on the robustness of the error estimate and step size selection procedures to handle derivative discontinuities. Codes that provide for event location and those based on control of the defect (residual) further improve the ability of a code to detect and resolve discontinuities. Because discontinuities in low-order derivatives are almost always present when solving DDEs, the error estimates are sometimes questionable. This is true even if a code goes to great pains to avoid stepping across discontinuities. For instance, `ddverk` monitors repeated step failures to discern whether the error estimate is not behaving as it ought for a smooth solution. If it finds a discontinuity in

this way, it uses special interpolants to get past the discontinuity. The solver generally handles discontinuities quite well since the defect does a good job of reflecting discontinuities. Similarly, `radar5` monitors error test failures to detect discontinuities and then treats the step size as a parameter to locate the discontinuity. Despite these precautions, the codes may still use questionable estimates near discontinuities. The `ddesd` solver takes a different approach. It exploits relationships between the residual and the error to obtain a plausible estimate of the error even when discontinuities are present; details may be found in [25].

### 9.3.3 Event Location

Just as with ODEs, it is often important to find out when something happens. For instance, we may need to find the first time a solution component attains a prescribed value because the problem changes then. Mathematically, this is formulated as finding a time  $t^*$  for which one of a collection of functions

$$g_1(t, y(t)), g_2(t, y(t)), \dots, g_k(t, y(t))$$

vanishes. We say that an *event* occurs at time  $t^*$  and the task is called *event location* [28]. As with locating discontinuities, the idea is to monitor the event functions for a change of sign between  $t_n$  and  $t_{n+1}$ . When a change is encountered in, say, equation  $m$ , the algebraic equation  $g_m(t, P(t)) = 0$  is solved for  $t^*$ . Here  $P(t)$  is the polynomial continuous extension that approximates  $y(t)$  on  $[t_n, t_{n+1}]$ . Event location is a valuable capability that we illustrate with a substantial example in Sect. 9.4. A few remarks about this example will illustrate some aspects of the task. A system of two differential equations is used to model a two-wheeled suitcase that may wobble from one wheel to the other. If the event  $y_1(t) - \pi/2 = 0$  occurs, the suitcase has fallen over and the computation comes to an end. If the event  $y_1(t) = 0$  occurs, a wheel has hit the floor. In this situation, we stop integrating and restart with initial conditions that account for the wheel bouncing. As this example makes clear, if there are events at all, we must find the *first* one if we are to model the physical situation properly. For both these event functions, the integration is to terminate at an event, but it is common that we want to know when an event occurs and the value of the solution at that time, but we want the integration to continue. A practical difficulty is illustrated by the event of a wheel bouncing – the integration is to restart with  $y_1(t) = 0$ , a terminal event! In the example we deal with this by using another capability, namely that we can tell the solver that we are interested only in events for which the function decreases through zero or increases through zero, or it does not matter how the function changes sign. Most DDE codes do not provide for *event location*. Among the codes that do are `dde23`, `ddesd`, and `dde_solver`.

### 9.3.4 Software Issues

A code needs values from the past, so it must use either a fixed mesh that matches the delays or some kind of continuous extension. The former approach, used in early codes, is impractical or impossible for most DDEs. Modern codes adopt the latter approach. Using some kind of interpolation, they evaluate the solution at delayed arguments, but this requires that they store all the information needed for the interpolation. This information is saved in a *solution history queue*. The older Fortran codes had available only static storage and using a static queue poses significant complications. Since the size of the queue is not known in advance, users must either allocate excessively large queues or live with the fact that the code will be unsuccessful for problems when the queue fills. The difficulty may be avoided to some extent by using circular queues in which the oldest solution is replaced by new information when necessary. Of course this approach fails if discarded information is needed later. The dynamic storage available in modern programming languages like MATLAB and Fortran 90/95 is vital to modern programs for the solution of DDEs. The `dde23`, `ddesd`, and `dde_solver` codes use dynamic memory allocation to make the management of the solution queue transparent to the user and to allow the solution queue to be used on return from the solver. Each of the codes trims the solution queue to the amount actually used at the end of the integration. The latest version of `dde_solver` has an option for trimming the solution queue during the integration while allowing the user to save the information conveniently if desired.

The more complex data structures available in modern languages are very helpful. The one used in the DDE solvers of MATLAB is called a *structure* and the equivalent in Fortran 90/95 is called a *derived type*. By encapsulating all information about the solution in a structure, the user is relieved of the details about how some things are accomplished. An example is evaluation of the solution anywhere in the interval of interest. The mesh and the details of how the solution is interpolated are unobtrusive when stored in a solution structure. A single function `deval` is used to evaluate the solution computed by any of the differential equation solvers of MATLAB. If the solution structure is called `sol`, there is a field, `sol.solver`, that is the name of the solver as a string, e.g., "dde23". With this `deval` knows how the data for the interpolant is stored and how to evaluate it. There are, in fact, a good many possibilities. For example, the ODE solvers that are based on linear multistep methods vary the order of the formula used from step to step, so stored in the structure is the order of the polynomial and the data defining it for each  $[t_n, t_{n+1}]$ . All the interpolants found in `deval` are polynomials, but several different representations are used because they are more natural to the various solvers. By encapsulating this information in a structure, the user need give no thought to the matter. A real dividend for libraries is that it is easy to add another solver to the collection.

The event location capability discussed in Sect. 9.3.3 requires output in addition to the solution itself, viz., the location of events, which event function led to each event reported, and the solution at each event. It is difficult to deal properly with event location without modern language capabilities because the number of events is not known in advance. In the ODE solvers of MATLAB, this information is available

in output arguments since the language provides for optional output arguments. Still, it is convenient to return the information as fields in a solution structure since a user may want to view only some of the fields. The equivalent in Fortran 90/95 is to return the solution as a derived type. This is especially convenient because the language does not provide for optional output. The example of §9.4.2 illustrates what a user interface might look like in both languages.

In the numerical example of §9.4.2, the solver returns after an event, changes the solution, and continues the integration. This is easy with an ODE because continuation can be treated as a new problem. Not so with DDEs because they need a history. Output as a structure is crucial to a convenient implementation of this capability. To continue an integration, the solver is called with the output structure from the prior integration instead of the usual history function or vector. The computation proceeds as usual, but approximate solutions at delayed arguments prior to the starting point are taken from the previously computed solution. If the delays depend on time and/or state, they might extend as far back as the initial data. This means that we must save the information needed to interpolate the solution from the initial point on. Indeed, the delays might be sufficiently long that values are taken from the history function or vector supplied for the first integration of the problem. This means that the history function or vector, as the case may be, must be held as a field in the solution structure for this purpose. A characteristic of the data structure is that fields do not have to be of the same type or size. Indeed, we have mentioned a field that is a string, fields that are arrays of length not known in advance, and a field that is a function handle.

## 9.4 Examples

Several collections of test problems are available to assess how well a particular DDE solver handles the issues and tasks described above. Each of the references [5, 7, 11, 19, 20, 25, 27, 29] describes a variety of test problems. In this section, the two problems of §9.1 are solved numerically to show that a modern DDE solver may be used to investigate complex problems. With such a solver, it is not much more difficult to solve a first-order system of DDEs than ODEs. Indeed, a design goal of the `dde23`, `ddesd`, and `dde_solver` codes was to exploit capabilities in MATLAB and Fortran 90/95 to make them as easy as possible to use, despite exceptional capabilities. We also supplement the discussion of event location in §9.3.3 with an example. Although it is easy enough to solve the DDEs, the problem changes at events and dealing with this is somewhat involved. We provide programs in both MATLAB and Fortran 90/95 that show even complex tasks can be solved conveniently with codes like `dde23` and `dde_solver`. Differences in the design of these codes are illustrated by this example. Further examples of the numerical solution of DDEs can be found in the documentation for the codes cited throughout the chapter. They can also be found in a number of the references cited and in particular, the references of Sect. 9.6.

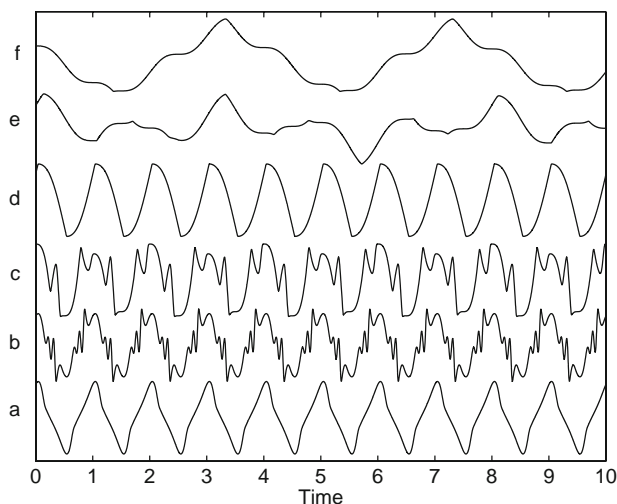
### 9.4.1 El-Niño Southern Oscillation Variability Model

Equation (9.2) is a DDE from [10] that models the El-Niño Southern Oscillation (ENSO) variability. It combines two key mechanisms that participate in ENSO dynamics, delayed negative feedback, and seasonal forcing. They suffice to generate very rich behavior that illustrates several important features of more detailed models and observational data sets. In [10] a stability analysis of the model is performed in a three-dimensional space of its strength of seasonal forcing  $b$ , atmosphere–ocean coupling  $\kappa$ , and propagation period  $\tau$  of oceanic waves across the Tropical Pacific. The physical parameters  $a, \kappa, \tau, b$ , and  $\omega$  are all real and positive.

Figure 9.3 depicts typical solutions computed with `dde_solver` and constant history  $h(t) = 1$  for  $t \leq 0$ . It shows six solutions obtained by fixing  $b = 1, \kappa = 100$  and varying the delay  $\tau$  over two orders of magnitude, from  $\tau = 10^{-2}$  to  $\tau = 1$ , with  $\tau$  increasing from bottom to top in the figure. The sequence of changes in solution type as  $\tau$  increases seen in this figure is typical for any choice of  $(b, \kappa)$ .

For a small delay,  $\tau < \pi/(2\kappa)$ , we have a periodic solution with period 1 (curve a); here the internal oscillator is completely dominated by the seasonal forcing. When the delay increases, the effect of the internal oscillator becomes visible: small wiggles, in the form of amplitude-modulated oscillations with a period of  $4\tau$ , emerge as the trajectory crosses the zero line. However, these wiggles do not affect the overall period, which is still 1. The wiggle amplitude grows with  $\tau$  (curve b) and eventually wins over the seasonal oscillations, resulting in period doubling (curve c). Further increase of  $\tau$  results in the model passing through a sequence of bifurcations that produce solution behavior of considerable interest for understanding ENSO variability.

Although solution of this DDE is straightforward with a modern solver, it is quite demanding for some parameters. For this reason, the compiled computation of the



**Fig. 9.3** Examples of DDE model solutions. Model parameters are  $\kappa = 100$  and  $b = 1$ , while  $\tau$  increases from curve (a) to curve (f) as follows: (a)  $\tau = 0.01$ , (b)  $\tau = 0.025$ , (c)  $\tau = 0.15$ , (d)  $\tau = 0.45$ , (e)  $\tau = 0.995$ , and (f)  $\tau = 1$

Fortran `dde_solver` was much more appropriate for the numerical study of [10] than the interpreted computation of the MATLAB `dde23`. The curves in Fig. 9.3 provide an indication of how much the behavior of the solution depends on the delay  $\tau$ . Further investigation of the solution behavior for different values of the other problem parameters requires the solution over extremely long intervals. The intervals are so long that it is impractical to retain all the information needed to evaluate an approximate solution anywhere in the interval. These problems led to the option of trimming the solution queue in `dde_solver`.

### 9.4.2 Rocking Suitcase

To illustrate event location for a DDE, we consider the following example from [26]. A two-wheeled suitcase may begin to rock from side to side as it is pulled. When this happens, the person pulling it attempts to return it to the vertical by applying a restoring moment to the handle. There is a delay in this response that can affect significantly the stability of the motion. This may be modeled with the DDE

$$\theta''(t) + \text{sign}(\theta(t))\gamma\cos(\theta(t)) - \sin(\theta(t)) + \beta\theta(t - \tau) = A\sin(\Omega t + \eta),$$

where  $\theta(t)$  is the angle of the suitcase to the vertical. This equation is solved on the interval  $[0, 12]$  as a pair of first-order equations with  $y_1(t) = \theta(t)$  and  $y_2(t) = \theta'(t)$ . Parameter values of interest are

$$\gamma = 2.48, \beta = 1, \tau = 0.1, A = 0.75, \Omega = 1.37, \eta = \arcsin\left(\frac{\gamma}{A}\right),$$

and the initial history is the constant vector zero. A wheel hits the ground (the suitcase is vertical) when  $y_1(t) = 0$ . The integration is then to be restarted with  $y_1(t) = 0$  and  $y_2(t)$  multiplied by the coefficient of restitution, here chosen to be 0.913. The suitcase is considered to have fallen over when  $|y_1(t)| = \frac{\pi}{2}$  and the run is then terminated. This problem is solved using `dde23` with the following MATLAB program.

```
function sol = suitcase
state = +1;
opts = ddeset('RelTol', 1e-5, 'Events', @events);
sol = dde23(@ddes, 0.1, [0; 0], [0 12], opts, state);

ref = [4.516757065, 9.751053145, 11.670393497];
fprintf('Kind of Event:           dde23 reference\n');
event = 0;
while sol.x(end) < 12
    event = event + 1;
    if sol.ie(end) == 1
        fprintf('A wheel hit the ground.
                %10.4f  %10.6f\n', ...
```

```

        sol.x(end), ref(event));
state = - state;
opts = ddeset(opts, 'InitialY', [ 0; 0.913*sol.y
    (2,end)]);
sol = dde23(@ddes,0.1,sol,[sol.x(end) 12],
    opts,state);
else
    fprintf('The suitcase fell over. %10.4f
        %10.6f\n',...
        sol.x(end),ref(event));
    break;
end
end
plot(sol.y(1,:),sol.y(2,:))
xlabel('\theta(t)')
ylabel('\theta''(t)')

%=====
function dydt = ddes(t,y,Z,state)
gamma = 0.248; beta = 1; A = 0.75; omega = 1.37;
ylag = Z(1,1);
dydt = [y(2); 0];
dydt(2) = sin(y(1)) - state*gamma*cos(y(1)) -
    beta*ylag ...
    + A*sin(omega*t + asin(gamma/A));

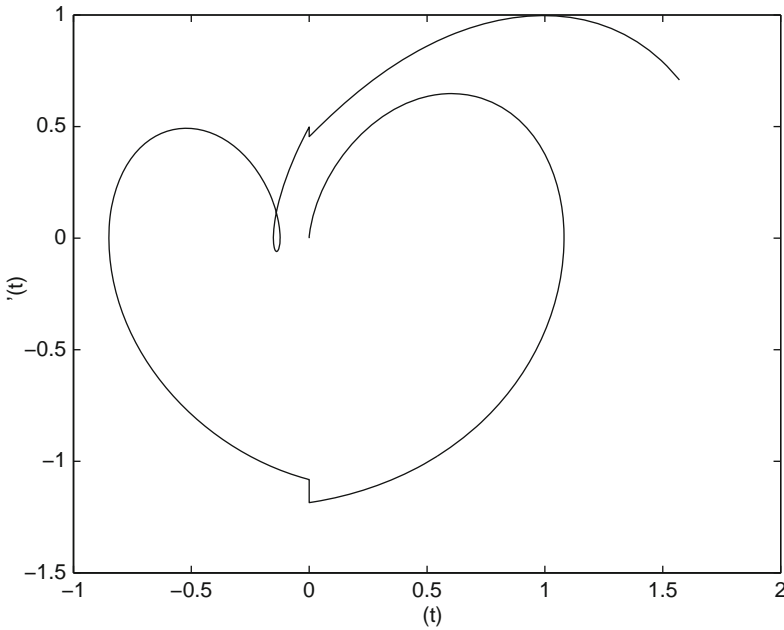
function [value,isterminal,direction] = events
    (t,y,Z,state)
value = [y(1); abs(y(1))-pi/2];
isterminal = [1; 1];
direction = [-state; 0];

```

The program produces the phase plane plot depicted in Fig. 9.4. It also reports what kind of event occurred and the location of the event. The reference values displayed were computed with the `dde_solver` code and much more stringent tolerances.

Kind of Event:	dde23	reference
A wheel hit the ground.	4.5168	4.516757
A wheel hit the ground.	9.7511	9.751053
The suitcase fell over.	11.6704	11.670393

This is a relatively complicated model, so we will elaborate on some aspects of the program. Coding of the DDE is straightforward except for evaluating properly the discontinuous coefficient  $\text{sign}(y_1(t))$ . This is accomplished by initializing a parameter `state` to `+1` and changing its sign whenever `dde23` returns because  $y_1(t)$  vanished. Handling `state` in this manner ensures that `dde23` does not need to deal



**Fig. 9.4** Two-wheeled suitcase problem

with the discontinuities it would otherwise see if the derivative were coded in a manner that allowed `state` to change before the integration is restarted; see [28] for a discussion of this issue. After a call to `dde23` we must consider why it has returned. One possibility is that it has reached the end of the interval of integration, as indicated by the last point reached, `sol.x(end)`, being equal to 12. Another is that the suitcase has fallen over, as indicated by `sol.ie(end)` being equal to 2. Both cases cause termination of the run. More interesting is a return because a wheel hit the ground,  $y_1(t) = 0$ , which is indicated by `sol.ie(end)` being equal to 1. The sign of `state` is then changed and the integration restarted. Because the wheel bounces, the solution at the end of the current integration, `sol.y(:,end)`, must be modified for use as initial value of the next integration. The `InitialY` option is used to deal with an initial value that is different from the history. The event  $y_1(t) = 0$  that terminates one integration occurs at the initial point of the next integration. As with the MATLAB IVP solvers, `dde23` does not terminate the run in this special situation of an event at the initial point. No special action is necessary, but the solver does locate and report an event at the initial point, so it is better practice to avoid this by defining more carefully the event function. When the indicator `state` is +1, respectively -1, we are interested in locating where the solution component  $y_1(t)$  vanishes only if it decreases, respectively increases, through zero. We inform the solver of this by setting the first component of the argument `direction` to -`state`. Notice that `dde23` is used to alter an existing options structure in the `while` loop. This is a convenient capability also present in `odeset`, the corresponding function for IVPs. The rest of the program is just a matter of reporting the results of the computations.

Default tolerances give an acceptable solution, though the phase plane plot would benefit from plotting more solution values. Reducing the relative error tolerance to  $1e-5$  gives better agreement with the reference values.

It is instructive to compare solving the problem with `dde23` to solving it with `dde_solver`. In contrast to the numerical study of the ENSO model, solving this problem is inexpensive in either computing environment, so the rather simpler program of the MATLAB program outweighs the advantage in speed of the Fortran program.

We begin by contrasting briefly the user interface of the solvers. A simple problem is solved simply by defining it with a call like

$$\text{SOL} = \text{DDE\_SOLVER}(\text{NVAR}, \text{DDES}, \text{BETA}, \text{HISTORY}, \text{TSPAN}) \quad (9.8)$$

Here `NVAR` is an integer array of two entries or three entries. The first is `NEQN`, the number of DDEs, the second is `NLAGS`, the number of delays, the third, if present, is `NEF`, the number of event functions. `DDES` is the name of a subroutine for evaluating the DDEs. It has the form

$$\text{SUBROUTINE DDES}(T, Y, Z, DY) \quad (9.9)$$

The input arguments are the independent variable  $T$ , a vector  $Y$  of `NEQN` components approximating  $y(T)$ , and an array  $Z$  that is `NEQN`  $\times$  `NLAGS`. Column  $j$  of this array is an approximation to  $y(\beta_j(T, y(T)))$ . The subroutine evaluates the DDEs with these arguments and returns  $y'(T)$  as the vector  $DY$  of `NEQN` components.

`BETA` is the name of a subroutine for evaluating the delays, and `HISTORY` is the name of a subroutine for evaluating the initial history function. More precisely, the functions are defined by subroutines for general problems, but they are defined in a simpler way in the very common situations of constant lags and/or constant history. `dde_solver` returns the numerical solution in the output structure `SOL`. The input vector `TSPAN` is used to inform the solver of the interval of integration and where approximate solutions are desired. `TSPAN` has at least two entries. The first entry is the initial point of the integration,  $t_0$ , and the last is the final point,  $t_f$ . If `TSPAN` has only two entries, approximate solutions are returned at all the mesh points selected by the solver itself. These points generally produce a smooth graph when the numerical solution is plotted. If `TSPAN` has entries  $t_0 < t_1 < \dots < t_f$ , the solver returns approximate solutions at (only) these points.

The call list of (9.8) resembles closely that of `dde23`. The design provides for a considerable variety of additional capabilities. This is accomplished in two ways. `F90` provides for optional arguments that can be supplied in any order if associated with a keyword. This is used, for example, to pass to the solver the name of a subroutine `EF` for evaluating event functions and the name of a subroutine `CHNG` in which necessary problem changes are made when event times are located, with a call like

$$\text{SOL} = \text{DDE\_SOLVER}(\text{NVAR}, \text{DDES}, \text{BETA}, \text{HISTORY}, \text{TSPAN}, \& \\ \text{EVENT\_FCN}=\text{EF}, \text{CHANGE\_FCN}=\text{CHNG})$$

This ability is precisely what is needed to solve this problem. EF is used to define the residuals  $g_1 = y_1$  and  $g_2 = |y_1| - \frac{\pi}{2}$ . CHNG is used to apply the coefficient of restitution and handle the STATE flag as in the solution for dde23. One of the optional arguments is a structure containing options. This structure is formed by a function called DDE.SET that is analogous to the function ddeset used by dde23 (and ddesd). The call list of (9.8) uses defaults for important quantities such as error tolerances, but of course, the user has the option of specifying quantities appropriate to the problem at hand. A more detailed discussion of these and other design issues may be found in [30]. Here is a Fortran 90 program that uses dde\_solver to solve this problem.

```

MODULE define_DDEs

  IMPLICIT NONE
  INTEGER, PARAMETER :: NEQN=2, NLAGS=1, NEF=2
  INTEGER :: STATE

CONTAINS

  SUBROUTINE DDES(T, Y, Z, DY)
    DOUBLE PRECISION :: T
    DOUBLE PRECISION, DIMENSION(NEQN) :: Y, DY
    DOUBLE PRECISION :: YLAG
    DOUBLE PRECISION, DIMENSION(NEQN,NLAGS) :: Z
    ! Physical parameters
    DOUBLE PRECISION, PARAMETER :: gamma=0.248D0,
                                   beta=1D0, &
                                   A=0.75D0,
                                   omega=1.37D0

    YLAG = Z(1,1)
    DY(1) = Y(2)
    DY(2) = SIN(Y(1)) - STATE*gamma*COS(Y(1)) -
           beta*YLAG &
           + A*SIN(omega*T + ASIN(gamma/A))

    RETURN
  END SUBROUTINE DDES

  SUBROUTINE EF(T, Y, DY, Z, G)
    DOUBLE PRECISION :: T
    DOUBLE PRECISION, DIMENSION(NEQN) :: Y, DY
    DOUBLE PRECISION, DIMENSION(NEQN,NLAGS) :: Z
    DOUBLE PRECISION, DIMENSION(NEF) :: G
    G = (/ Y(1), ABS(Y(1)) - ASIN(1D0) /)
    RETURN
  END SUBROUTINE EF

```

```

SUBROUTINE CHNG(NEVENT, TEVENT, YEVENT, DYEVENT,
                HINIT, &
                DIRECTION, ISTERMINAL, QUIT)
  INTEGER :: NEVENT
  INTEGER, DIMENSION(NEF) :: DIRECTION
  DOUBLE PRECISION :: TEVENT, HINIT
  DOUBLE PRECISION, DIMENSION(NEQN) :: YEVENT,
    DYEVENT
  LOGICAL :: QUIT
  LOGICAL, DIMENSION(NEF) :: ISTERMINAL
  INTENT(IN) :: NEVENT, TEVENT
  INTENT(INOUT) :: YEVENT, DYEVENT, HINIT,
    DIRECTION, &
    ISTERMINAL, QUIT
  IF (NEVENT == 1) THEN
    ! Restart the integration with initial values
    ! that correspond to a bounce of the suitcase.
    STATE = -STATE
    YEVENT(1) = 0.0D0
    YEVENT(2) = 0.913*YEVENT(2)
    DIRECTION(1) = - DIRECTION(1)
  ! ELSE
  !   Note:
  !   The suitcase fell over, NEVENT = 2.
  !       The integration
  !   could be terminated by QUIT = .TRUE.,
  !       but this
  !   event is already a terminal event.
  ENENDIF
  RETURN
END SUBROUTINE CHNG

END MODULE define_DDEs

!*****

PROGRAM suitcase

! The DDE is defined in the module define_DDEs. The
! problem
! is solved here with ddd_solver and its output
! written to
! a file. The auxiliary function suitcase.m imports
! the data
! into Matlab and plots it.

```

```

USE define_DDEs
USE DDE_SOLVER_M

IMPLICIT NONE

! The quantities
! NEQN = number of equations
! NLAGS = number of delays
! NEF = number of event functions
! are defined in the module define_DDEs as
! PARAMETERS so
! they can be used for dimensioning arrays here.
! They ! pre assed to the solver in the array NVAR.
! INTEGER, DIMENSION(3) :: NVAR = (/NEQN,NLAGS,NEF/)

TYPE(DDE_SOL) :: SOL
! The fields of SOL are expressed in terms of the
! number of differential equations, NEQN, and the
! number of output points, NPTS:
! SOL%NPTS -- NPTS, number of output points.
! SOL%T(NPTS) -- values of independent
!               variable, T.
! SOL%Y(NPTS,NEQN) -- values of dependent
!                   variable, Y,
!                   corresponding to values of
!                   SOL%T.
! When there is an event function, there are fields
! SOL%NE -- NE, number of events.
! SOL%TE(NE) -- locations of events
! SOL%YE(NE,NEQN) -- values of solution at events
! SOL%IE(NE) -- identifies which event
!               occurred

TYPE(DDE_OPTS) :: OPTS

! Local variables:
INTEGER :: I,J

! Prepare output points.
INTEGER, PARAMETER :: NOUT=1000
DOUBLE PRECISION, PARAMETER :: T0=0D0,TFINAL=12D0
DOUBLE PRECISION, DIMENSION(NOUT) :: TSPAN= &
(/ (T0+(I-1)*((TFINAL-T0)/(NOUT-1))), I=1,NOUT)/)

```

```

! Initialize the global variable that governs the
! form of the DDEs.

STATE = 1
! Set desired integration options.
OPTS = DDE_SET(RE=1D-5,DIRECTION=(/-1,0/), &
              ISTERMINAL=(/ .FALSE., .TRUE. /))

! Perform the integration.
SOL = DDE_SOLVER(NVAR,DDES,(/0.1D0/),(/0D0,0D0/), &
                TSPAN,OPTIONS=OPTS,EVENT_FCN=EF,CHANGE_FCN=
                CHNG)

! Was the solver successful?
IF (SOL%FLAG == 0) THEN
    ! Write the solution to a file for subsequent
    ! plotting in Matlab.
    OPEN(UNIT=6, FILE='suitcase.dat')
    DO I = 1,SOL%NPTS
        WRITE(UNIT=6,FMT='(3D12.4)') SOL%T(I), &
                                           (SOL%Y(I,J),J=1,
                                           NEQN)
    ENDDO
    PRINT *, ' Normal return from DDE_SOLVER with
              results'
    PRINT *, " written to the file 'suitcase.dat'."
    PRINT *, ' '
    PRINT *, ' These results can be accessed in
    PRINT *, ' Matlab' and plotted in a phase plane by'
    PRINT *, ' '
    PRINT *, " >> [t,y] = suitcase;"
    PRINT *, ' '
    PRINT *, ' '
    PRINT *, ' Kind of Event:'
    DO I = 1,SOL%NE
        IF(SOL%IE(I) == 1) THEN
            PRINT *, ' A wheel hit the ground at',
                    SOL%TE(I)
        ELSE
            PRINT *, ' The suitcase fell over at',
                    SOL%TE(I)
        END IF
    END DO
    PRINT *, ' '
ELSE

```

```

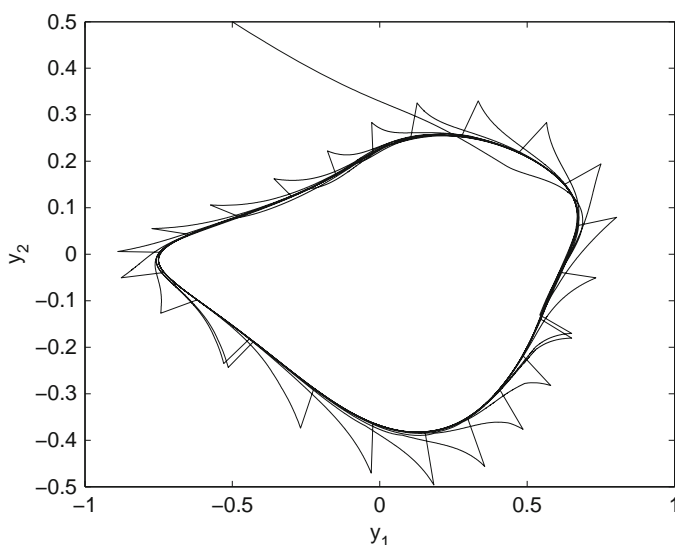
PRINT *, ' Abnormal return from DDE_SOLVER.
      FLAG = ', &
      SOL%FLAG
ENDIF

STOP
END PROGRAM suitcase

```

### 9.4.3 Time-Dependent DDE with Impulses

In Sect. 9.1 we stated a first-order system of DDEs that arises in modeling cellular neural networks [31] and commented that it is relatively difficult to solve numerically because the delays vanish periodically during the integration. It is also difficult because the system is subject to impulse loading. Specifically, at each  $t_k = 2k$  an impulse is applied by replacing  $y_1(t_k)$  with  $1.2y_1(t_k)$  and  $y_2(t_k)$  by  $1.3y_2(t_k)$ . Because the impulses are applied at specific times, this problem might be solved in several ways. When using `dde_solver`, it is convenient to define an event function  $g(t) = t - T_e$ , where  $T_e = 2$  initially and  $T_e = 2(k+1)$  once  $T = 2k$  has located. This could be done with `ddesd`, too, but the design of the MATLAB solver makes it more natural simply to integrate to  $t_k$ , return to the calling program where the solution is altered because of the impulse, and call the solver to continue the integration.



**Fig. 9.5** Neural network DDE with time-dependent impulses

This is much like the computation of the suitcase example. Figure 9.5 shows the phase plane for the solution computed with impulses using `dde_solver`. Similar results were obtained with `ddesd`.

## 9.5 Conclusion

We believe that the use of models based on DDEs has been hindered by the availability of quality software. Indeed, the first mathematical software for this purpose [17] appeared in 1975. As software has become available that makes it not greatly harder to integrate differential equations with delays, there has been a substantial and growing interest in DDE models in all areas of science. Although the software benefited significantly from advances in ODE software technology, we have seen in this chapter that there are numerical difficulties and issues peculiar to DDEs that must be considered. As interest has grown in DDE models, so has interest in both developing algorithms and quality software. There are classes of problems that remain challenging, but the examples of Sect. 9.4 show that it is not hard to use quality DDE solvers to solve realistic and complex DDE models.

## 9.6 Further Reading

A short, but good list of sources is

1. Baker C T H, Paul C A H, and Willé D R [2], A Bibliography on the Numerical Solution of Delay Differential Equations
2. Bellen A and Zennaro M [3], Numerical Methods for Delay Differential Equations
3. Shampine L F, Gladwell I, and Thompson S [26], Solving ODEs with MATLAB
4. <http://www.radford.edu/~thompson/ffddes/index.html>, a web site devoted to DDEs

In addition, the numerical analysis section of Scholarpedia [23] contains several very readable general articles devoted to the numerical solution of delay differential equations.

## References

1. Baker C T H and Paul C A H (1996), A Global Convergence Theorem for a Class of Parallel Continuous Explicit Runge–Kutta Methods and Vanishing Lag Delay Differential Equations, SIAM J Numer Anal 33:1559–1576

2. Baker C T H, Paul C A H, and Willé D R (1995), A Bibliography on the Numerical Solution of Delay Differential Equations, Numerical Analysis Report 269, Mathematics Department, University of Manchester, U.K.
3. Bellen A and Zennaro M (2003), Numerical Methods for Delay Differential Equations, Oxford Science, Clarendon Press
4. Bogacki P and Shampine L F (1989), A 3(2) Pair of Runge–Kutta Formulas, *Appl Math Lett* 2:1–9
5. Corwin S P, Sarafyan D, and Thompson S (1997), DKL6G: A Code Based on Continuously Imbedded Sixth Order Runge–Kutta Methods for the Solution of State Dependent Functional Differential Equations, *Appl Numer Math* 24:319–333
6. Corwin S P, Thompson S, and White S M (2008), Solving ODEs and DDEs with Impulses, *JNAIAM* 3:139–149
7. Enright W H and Hayashi H (1997), A Delay Differential Equation Solver Based on a Continuous Runge–Kutta Method with Defect Control, *Numer Alg* 16:349–364
8. Enright W H and Hayashi H (1998), Convergence Analysis of the Solution of Retarded and Neutral Differential Equations by Continuous Methods, *SIAM J Numer Anal* 35:572–585
9. El'sgol'ts L E and Norkin S B (1973), Introduction to the Theory and Application of Differential Equations with Deviating Arguments, Academic Press, New York
10. Ghil M, Zaliapin I, and Thompson S (2008), A Differential Delay Model of ENSO Variability: Parametric Instability and the Distribution of Extremes, *Nonlin Processes Geophys* 15:417–433
11. Guglielmi N and Hairer E (2008), Computing Breaking Points in Implicit Delay Differential Equations, *Adv Comput Math*
12. Guglielmi N and Hairer E (2001), Implementing Radau IIA Methods for Stiff Delay Differential Equations, *Computing* 67:1–12
13. Hairer E, Nørsett S P, and Wanner G (1987), Solving Ordinary Differential Equations I, Springer–Verlag, Berlin, Germany
14. Jackiewicz Z (2002), Implementation of DIMSIMs for Stiff Differential Systems, *Appl Numer Math* 42:251–267
15. Jackiewicz Z and Lo E (2006), Numerical Solution of Neutral Functional Differential Equations by Adams Methods in Divided Difference Form, *J Comput Appl Math* 189:592–605
16. MATLAB 7 (2006), The MathWorks, Inc., 3 Apple Hill Dr., Natick, MA 01760
17. Neves K W (1975), Automatic Integration of Functional Differential Equations: An Approach, *ACM Trans Math Softw* 1:357–368
18. Neves K W and Feldstein A (1976), Characterization of Jump Discontinuities for State Dependent Delay Differential Equations, *J Math Anal Appl* 56:689–707
19. Neves K W and Thompson S (1992), Software for the Numerical Solution of Systems of Functional Differential Equations with State Dependent Delays, *Appl Numer Math* 9:385–401
20. Paul C A H (1994), A Test Set of Functional Differential Equations, Numerical Analysis Report 243, Mathematics Department, University of Manchester, U.K.
21. Paul C A H (1995), A User–Guide to ARCHI, Numerical Analysis Report 283, Mathematics Department, University of Manchester, U.K.
22. Paul C A H (1992), Developing a Delay Differential Equation Solver, *Appl Numer Math* 9:403–414
23. Scholarpedia, <http://www.scholarpedia.org/>
24. Shampine L F (1994), Numerical Solution of Ordinary Differential Equations, Chapman & Hall, New York
25. Shampine L F (2005), Solving ODEs and DDEs with Residual Control, *Appl Numer Math* 52:113–127
26. Shampine L F, Gladwell I, and Thompson S (2003) Solving ODEs with MATLAB, Cambridge Univ. Press, New York
27. Shampine L F and Thompson S, Web Support Page for dde\_solver, <http://www.radford.edu/~thompson/ffddes/index.html>

28. Shampine L F and Thompson S (2000), Event Location for Ordinary Differential Equations, *Comp Maths Appls* 39:43–54
29. Shampine L F and Thompson S (2001), Solving DDEs in MATLAB, *Appl Numer Math* 37:441–458
30. Thompson S and Shampine L F (2006), A Friendly Fortran DDE Solver, *Appl Numer Math* 56:503–516
31. Yongqing Y and Cao J (2007), Stability and Periodicity in Delayed Cellular Neural Networks with Impulsive Effects, *Nonlinear Anal Real World Appl* 8:362–374

# Chapter 10

## Effects of Time Delay on Synchronization and Firing Patterns in Coupled Neuronal Systems

Qishao Lu, Qingyun Wang, and Xia Shi

**Abstract** Synchronization and firing patterns of neurons play a significant role in neural signal encoding and transduction of information processing of nervous systems. For real neurons, the signal transmission delay is inherent due to finite propagation speed of action potentials through axons, as well as both dendritic and synaptic processing. Time delay is universal and inevitable, and results in complex dynamical behavior, such as multistability and bifurcations leading to chaos. In this chapter, the authors provide a survey of some recent developments on the effects of time delay on synchronization and firing patterns in coupled neuronal systems. Basic concepts of firing patterns of neurons and complete and phase synchronization of oscillators are introduced. Synchronization and firing patterns in electrically coupled neurons as well as inhibitory and excitatory delayed synapses with time delay are presented. Delay effects on spatiotemporal dynamics of coupled neuronal activities are discussed. Some well-known neuron models are also included for benefit of the readers.

**Keywords:** Firing pattern · Time delay · Neuronal system · Synchronization

### 10.1 Introduction

Neurons are the basic building blocks of nervous systems, and there are about  $10^{11}$  neurons in the human brain. These specialized cells are the information processing units of the brain responsible for receiving and transmitting information. Neurons are coupled in an extremely complex neural network, and neural information is mainly encoded and integrated through various firing patterns of neurons [1–4]. To understand the essence of collective behavior of coupled neurons, various firing patterns of coupled neurons should be studied theoretically and numerically by means of neural models.

Synchronization of neurons plays a significant role in neural signal encoding and transduction of information processing of nervous systems. Physiological experiments have indicated the existence of synchronous motions of neurons in different areas of the brain. Quasiperiodic synchronous rhythm was shown in cortex and a small neural systems like central pattern generators. Synchronous firing of neurons was also observed in cat visual cortex. The observations of synchronous neural activity in the central nervous system have stimulated a great deal of theoretical work on synchronization in coupled neural networks.

Electrophysiological and anatomical data indicate that the cerebral cortex is spatially and functionally organized. These patterns of connectivity must influence the intrinsic dynamics of cortical circuits which can now be visualized using optical techniques. The relationship between the spatial profile of neural interactions and spatiotemporal patterns of neuronal activity can be investigated in modeling studies.

An important property of neural interactions is that they involve delays. We know that the information flow in neural systems is not instantaneous in general. The signal transmission delay is inherent due to finite propagation speed of action potentials through axons, as well as both dendritic and synaptic processing. For example, conduction velocity ranges within  $10 \text{ m s}^{-1}$ , leading to nonnegligible transmission times from milliseconds to hundreds milliseconds for propagation through the cortical network. Effective delays can also be induced by the spike generation dynamics. Hence, for real neurons the effect of time delay is universal and inevitable in the behavior of synchronization and firing patterns of coupled neuronal systems. Behaviors due to time delays, such as multistability and bifurcation leading to chaos, have been analytically and numerically investigated by using some simple neuronal models (such as integrate-and-fire (IF), FitzHugh-Nagumo (FHN), Hindmarsh-Rose (HR), and Hodgkin-Huxley (HH) models) with delayed coupling.

The effect of time delays on the coupled oscillators was studied extensively [5–9]. Recently, the synchronization dynamics for a system of two HH neurons with delayed diffusive and pulsed coupling was investigated in [10]. The stability of a network of coupled HH neurons was analyzed and the stability regions in the parameter space were obtained. With diffusive coupling, there are three regions in the parameter space, corresponding to qualitatively distinct behavior of the coupled dynamics. In particular, the two neurons can synchronize in the two regions and desynchronize in the third. Reference [11] studied enhancement of neural synchrony by time delay, that is, a stable synchronized state existing at low coupling strengths for significant time delays, in a time-delayed system of two chaotic HR neurons. It was shown that in the delayed system there was always an extended region of stable synchronous activity corresponding to low coupling strengths, which could be achieved only by much higher coupling strengths without delay. Bursting behavior in a pair of HR neurons with delayed coupling was studied in [12], in which bifurcations due to time lag and coupling as well as the stability of stationary state corresponding to the quiescence behavior were analyzed. It is found that bursting is created by coupling and its properties strongly depended on the time lag. In particular, there is a domain of values of time lags which render the bursting of the two neurons exactly synchronous. Moreover, phenomena of suppression of oscillations and synchronization

are important in understanding the physiology of neuronal systems, and have been intensively studied. In [13], a method for suppression of synchrony in a globally coupled oscillator network was suggested based on time-delayed feedback via the mean field. A theory was developed based on the consideration of the synchronization transition as a Hopf bifurcation. Also time-delay-induced switching between in phase and antiphase oscillations has been observed in several systems [8, 9].

When interactions are spatially structured, delays can induce a wealth of dynamical states with different spatiotemporal properties and domains of multistability. In [14] for the dynamics of large networks of neurons, it was shown that delays gave rise to a wealth of bifurcations and to a rich phase diagram, which includes oscillatory bumps, traveling waves, lurching waves, standing waves arising via a period-doubling bifurcation, aperiodic regimes, and regimes of multistability. The existence and the stability of various dynamical patterns was studied analytically and numerically in a simplified firing-rate model as a function of the interaction parameters.

As we know, noise could have quite different qualitative effects on the deterministic dynamics depending on the values of time lag and coupling. It has been shown elsewhere that such delays may lead to homogeneous oscillations in neural networks with random connectivity. Dynamics of FHN neuron ensembles with time-delayed couplings subject to white noise, was studied by using both direct simulations and a semianalytical augmented moment method in [15]. Effects of the parameters of coupling strength, delay, noise intensity, and the ensemble size on the emergence of the oscillation and on the synchronization in FN neuron ensembles were studied. The synchronization showed the fluctuation-induced enhancement at the transition between nonoscillating and oscillating states. Reference [16] explored the dynamics of a HH-type model for thermally sensitive neurons with time-delayed feedback that exhibit intrinsic oscillatory activity. The dynamics of the neuron depending on the temperature, the synaptic strength, and the delay time was investigated. The parameter regions where the effect of the recurrent connection is excitatory, inducing spikes or trains of spikes, and the regions where it is inhibitory, reducing or eliminating completely the spiking behavior, were found. The complex interplay of the intrinsic dynamics of the neuron with the recurrent feedback input and a noisy input was revealed.

The influence of white noise on the dynamics of a pair of stochastically perturbed HR bursting neurons with delayed electrical coupling was studied in [17, 18]. Possibility of stochastically stable exact synchronization with sufficiently strong coupling was proved for arbitrary time lags and sufficiently small noise. In particular, a simple method to predict the intensity of noise that can destabilize the quiescent state was proposed and compared with numerical computations. Furthermore, it was demonstrated that quite small noise might completely destroy the exact synchronization of bursting dynamics.

In this chapter, we make surveys on some recent developments in the effects of time delay on synchronization and firing patterns in coupled neuronal systems. In Sect. 10.2, basic concepts of firing patterns of neurons and complete and phase synchronization of oscillators are introduced. Section 10.3 contains a discussion of synchronization and firing patterns in electrically coupled neurons with time delay.

Synchronization and firing patterns in coupled neurons with inhibitory and excitatory delayed synapses are presented in Sects. 10.4 and 10.5, respectively. Delay effects on spatiotemporal dynamics of coupled neuronal activity are discussed by the authors in Sect. 10.6. Concluding remarks are given in Sect. 10.7, and some well-known neuron models are introduced in the Appendices for the convenience of the reader.

## 10.2 Basic Concepts

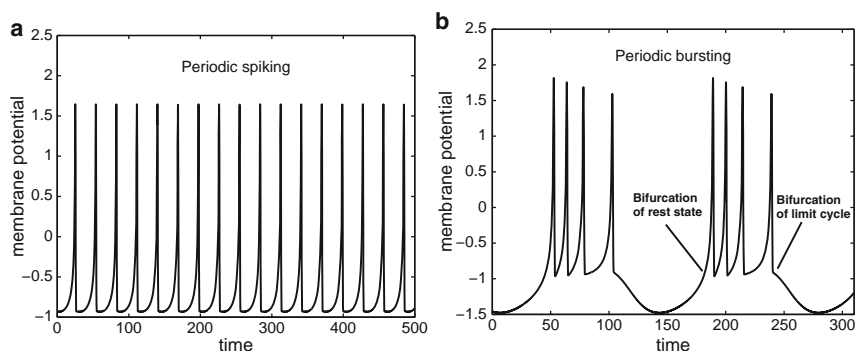
### *10.2.1 Firing Patterns of a Single Neuron*

A neuron consists of the dendrites, the soma, and the axon. Each part of the neuron plays a role in the communication of information throughout the body. The dendrites receive information from other neurons and transmit electrical stimulation to the soma, and the signals are joined in the soma and passed on to the axon, which transmits the neural signal to the other neurons.

The junction between two neurons is called a synapse. It is common to refer to the sending neuron as the presynaptic cell and to the receiving neuron as the postsynaptic cell. A single neuron in vertebrate cortex often connects to more than  $10^4$  postsynaptic neurons. There are two common types of synapses in the vertebrate brain, that is, electrical and chemical synapses. According to these junctions, the couplings among neurons are divided into electrical and chemical ones in coupled neuronal systems. Furthermore, the chemical coupling is also classified either as excitatory or inhibitory one according to its function.

Neural signals are expressed by the action potentials, which are also called spikes or impulses. Action potentials are generated and sustained by ionic currents through the cell membrane. A neuron receives inputs from other neurons through the synapses and the inputs produce electrical transmembrane currents that change the membrane potential of the neuron. The synaptic currents produce postsynaptic potentials that could be amplified by the voltage-sensitive channels embedded in neuronal membrane and lead to the generation of action potentials or spikes, that is, abrupt and transient changes of membrane voltage that propagate to other neurons via the axon. The dynamical models illustrating the generation of spikes in a neuron are presented in Appendices.

A neuron is quiescent if its membrane potential is at rest or exhibits small amplitude (“subthreshold”) oscillations. In the point of dynamics, this corresponds to the system residing at an equilibrium or a small amplitude limit cycle attractor, respectively. A neuron is said to be excitable if a small perturbation away from a quiescent state can result in a large excursion of its action potential before returning to quiescence. Then the neuron can fire spikes when there is a large amplitude limit cycle attractor, which may coexist with the quiescent state. Neuronal firing is crucial to the information processing in the nervous system, and there are many complex firing patterns observed in neural experiments and numerical simulations.



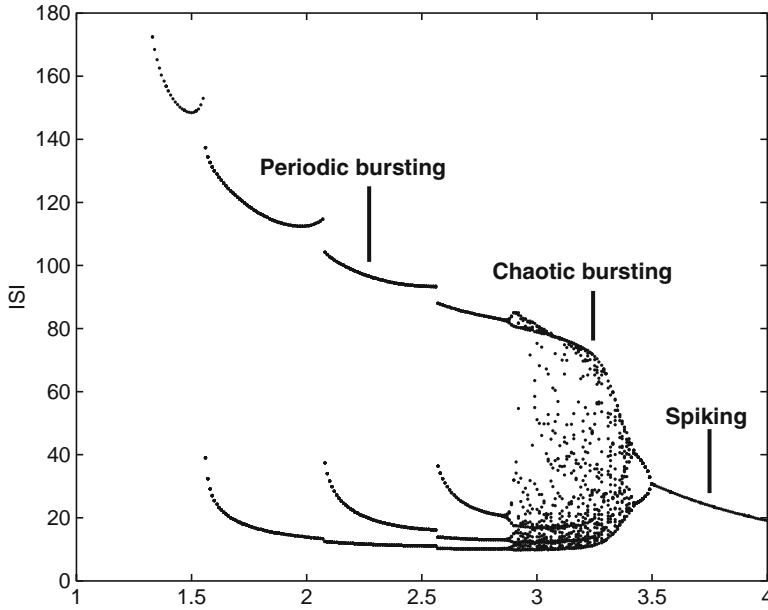
**Fig. 10.1** Neuronal spiking and bursting

There are two basic types of neuronal firing patterns, that is, spiking and bursting. Repetitive spiking comes from a large amplitude limit cycle, which corresponds to oscillatory dynamics of neurons, Fig. 10.1a. While the bursting depends on two important bifurcations, that is, the bifurcation of a quiescent state that leads to repetitive spiking and the bifurcation of a spiking attractor that results in quiescence, as shown in Fig. 10.1b. Bursting appears when the dynamical neural system contains two timescales and the slowly changing variable visits the spiking and quiescent states alternately. The neuron may fire several spikes in the duration of each burst, depending on the period of the slowly changing variable spending in the spiking area relative to the period of a single spike. The classification of bursting was presented by Izhikevich [19] according to the bifurcations between the quiescent state and that of the repetitive spiking. The bifurcation scenarios of the interspike interval (ISI) series can often be used to identify the bursting types by the number of spikes per burst and to show whether the bursting pattern is periodic or chaotic (Fig. 10.2).

## 10.2.2 Synchronization

In a classical context, synchronization means adjustment of rhythms of self-sustained periodic oscillators due to weak interaction and can be described in terms of phase locking and frequency entrainment. Nowadays, it is well-known that self-sustained oscillators can generate rather complex, chaotic signals. Recent studies have revealed that such systems, being coupled, are able to undergo synchronization. Therefore, modern concept of synchronization also covers coupled chaotic systems. Chaotic synchronization can be defined as the adjustment of varying rhythms of coupled chaotic oscillators. One distinguishes synchronization in different forms, that is, complete, phase, lag, and generalized synchronization, etc.

Complete synchronization of coupled identical systems means the coincidence of state variables with time evolution, starting from different initial points in the state space. This kind of synchronization usually can be reached by strong coupling. Let



**Fig. 10.2** Some bifurcation scenarios of ISI of neurons

us begin with a general system with temporal evolution governed by the following equation.

$$\dot{\mathbf{w}} = f(\mathbf{w}).$$

Here  $\mathbf{w} = (w_1, w_2, \dots, w_n)$  is an  $n$ -dimensional state vector,  $f : R^n \rightarrow R^n$  is a vector function. Through a bidirectional coupling, we obtain a coupled system as follows:

$$\begin{aligned}\dot{\mathbf{v}} &= f(\mathbf{v}) + C(\mathbf{w} - \mathbf{v}), \\ \dot{\mathbf{w}} &= f(\mathbf{w}) + C(\mathbf{v} - \mathbf{w}),\end{aligned}\tag{10.1}$$

in which  $C$  is the coupling matrix. For the coupled neurons, this would mimic the electrical coupling. In this framework, complete synchronization is defined as the identity between the trajectories of two systems. Then the existence of complete synchronization requires that the synchronization manifold  $\mathbf{v} \equiv \mathbf{w}$  is asymptotically stable, or equivalently,  $\lim_{t \rightarrow \infty} \mathbf{e}(t) = 0$ , where  $\mathbf{e}(t)$  is the synchronization error defined by  $\mathbf{e}(t) \equiv \mathbf{v} - \mathbf{w}$ . This property can be justified in general by using the stability analysis of the following linearized system for small  $\mathbf{e}$ :

$$\begin{aligned}\dot{\mathbf{e}} &= f(\mathbf{w}) - f(\mathbf{v}) + C(\mathbf{v} - \mathbf{w}) - C(\mathbf{w} - \mathbf{v}) \\ &= (Df(\mathbf{w}) + 2C)\mathbf{e},\end{aligned}\tag{10.2}$$

where  $Df(\mathbf{w})$  is the Jacobian matrix of the vector field  $f$  evaluated onto the trajectory of any one of the coupled systems.

Let  $J = Df(\mathbf{w}) + 2C$ . Three types of complete synchronization will be considered theoretically as follows. First, if the trajectory of  $\mathbf{w}(t)$  is a fixed point, then the study of the stability problem of system (10.2) can be made by evaluating the eigenvalues of the matrix  $J$ . Therefore, a condition for the complete synchronization of fixed points of the coupled system (10.1) is that the real parts of all eigenvalues of  $J$  are smaller than zero. Second, when the trajectory of  $\mathbf{w}(t)$  is periodic, (10.2) is a linear ordinary differential equation with periodic coefficients. According to the Floquet theorem, the stability problem of (10.1) can be solved through calculating the Floquet multipliers. A condition for asymptotic stability of the trivial solution of (10.1), that is, a condition for the complete synchronization of periodic motions of the coupled system (10.1) is that the modula of all characteristic multipliers are not larger than 1. Finally, if the trajectory of  $\mathbf{w}(t)$  is chaotic, then we need to calculate the Lyapunov exponents of system (10.2), that is, the conditional Lyapunov exponents. A condition for the complete synchronization of chaotic motions of the coupled system (10.1) is that all conditional Lyapunov exponents are smaller than zero.

If the coupling is weak, another kind of synchronization, phase synchronization, can be reached for two coupled nonidentical systems. Many physical, chemical, and biological systems can produce self-sustained oscillations, which correspond mathematically to a stable limit cycle in the state space of an autonomous continuous-time dynamical system, and the phase  $\phi$  can be introduced as the variable parameterizing the motion along this cycle. Usually, the phase can be chosen in such a way that it grows uniformly in time, that is,  $d\phi/dt = \omega$ , where  $\omega$  is the natural frequency of oscillations. Then for two coupled periodic oscillators,  $n:m$  phase locking or frequency locking is represented by  $|n\phi_1 - m\phi_2| < \text{constant}$  or  $n\omega_1 = m\omega_2$ .

For phase synchronization of coupled chaotic oscillators, the primary problem is to determine the time-dependent phase  $\phi(t)$ . A few approaches have been proposed [20] to calculate the phases of chaotic oscillators. While for chaotic neuronal systems, the Poincaré section-based method is usually used, in which the rotation angle of a radius vector projection on a certain plane of variables is defined to calculate the instantaneous phase.

Choose a Poincaré section defined as  $\Sigma$ . A trajectory of the system successively intersects  $\Sigma$  and each intersection is associated with a phase increase of  $2\pi$ . Then the phase can be defined by

$$\phi(t) = 2\pi \frac{t - T_i}{T_{i+1} - T_i} + 2\pi i \quad (T_i < t < T_{i+1}) \quad (10.3)$$

where  $t$  is the time and  $T_i$  is the time of the  $i$ th intersection of the trajectory with the Poincaré section. With this definition of phase, the mean frequency is

$$\langle \omega \rangle = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \frac{2\pi}{T_{i+1} - T_i}.$$

Similar to phase locking of coupled periodic oscillators, the condition of phase synchronization of two coupled chaotic oscillators can be formalized as  $|m\phi_1(t) - n\phi_2(t)| < \text{const}$ , where  $m$  and  $n$  are integers. Phase synchronization is also defined

as frequency entrainment, provided that the mean frequencies of the oscillators are in rational relation, that is  $|m\omega_1 - n\omega_2| = 0$ , where  $m$  and  $n$  are same as the above description. In particular, it is called in-phase synchronization when  $|\phi_1(t) - \phi_2(t)| = 0$  and antiphase synchronization when  $|\phi_1(t) - \phi_2(t)| = \pi$ . For more results about weak coupling, one can refer to the book [21].

### 10.3 Synchronization and Firing Patterns in Electrically Coupled Neurons with Time Delay

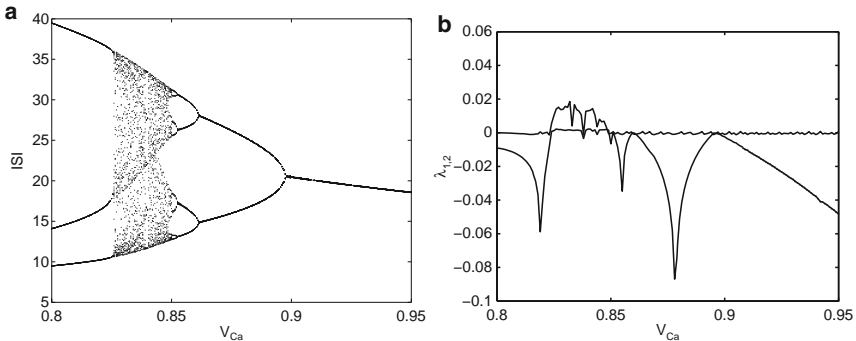
In [22], we consider the synchronization and firing patterns of two electrically coupled ML neurons with time delay. For a ML neuron (see Appendix 10.2),  $V_{Ca}$  is considered as a control parameter. To generate rich firing patterns in the single ML neuron model, an external linear slow subsystem is added as follows:

$$\frac{dI}{dt} = \mu(0.2 + V), \quad (10.4)$$

where  $\mu$  is a constant, which takes  $\mu = 0.005$  below. When the parameter  $V_{Ca}$  changes, the ML neuron model can exhibit rich firing behavior, such as various periodic and chaotic patterns. The bifurcation diagram of ISI and the two largest Lyapunov exponents are shown in Fig. 10.3, in which various firing behavior of a single ML neuron is exhibited with the parameter  $V_{Ca}$  changing.

Two coupled ML neurons with a time delayed gap junction are described by the following equations:

$$\begin{aligned} \frac{dV_{1,2}}{dt} = & g_{Ca}m_{\infty}(V_{Ca} - V_{1,2}) + g_k\omega(V_k - V_{1,2}) +, \\ & g_l(V_l - V_{1,2}) - I_{1,2} + C(V_{2,1}(t - \tau) - V_{1,2}), \end{aligned}$$



**Fig. 10.3** **a** The bifurcation diagram of ISI vs. the parameter  $V_{Ca}$  in the ML neuron model; **b** the corresponding two largest Lyapunov exponents  $\lambda_{1,2}$  of **a**

$$\begin{aligned}\frac{d\omega_{1,2}}{dt} &= \lambda_{\infty}(V_{1,2})(\omega_{\infty}(V_{1,2}) - \omega_{1,2}), \\ \frac{dI_{1,2}}{dt} &= \mu(0.2 + V_{1,2}),\end{aligned}\quad (10.5)$$

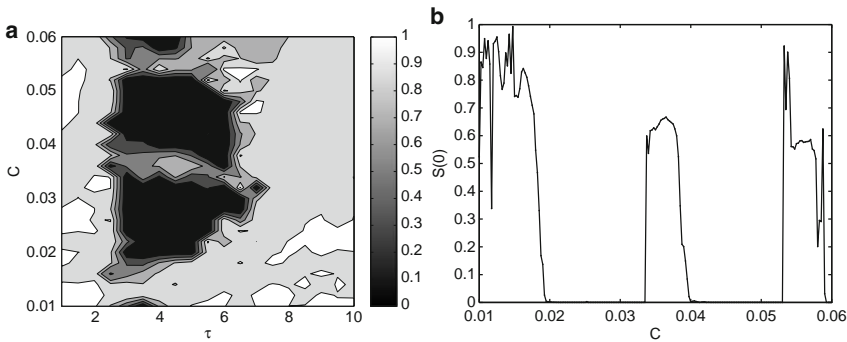
where  $C$  is the coupling strength and  $\tau$  is the time delay in gap junction, the subscript 1 (or 2) represents the neuron 1 (or 2). Choose the parameter  $V_{Ca} = 0.845$  mV, at which a single ML neuron exhibits chaotic firing behavior as shown in Fig. 10.3.

It is shown in [22] that when the coupling strength is below  $C = 0.6$ , the coupled neurons cannot realize synchronization. In what follows, we focus on the effect of time-delay coupling on synchronization and firing patterns of two coupled chaotic ML neurons. To investigate the synchronization of two coupled neurons with time delay, the similarity function is introduced as

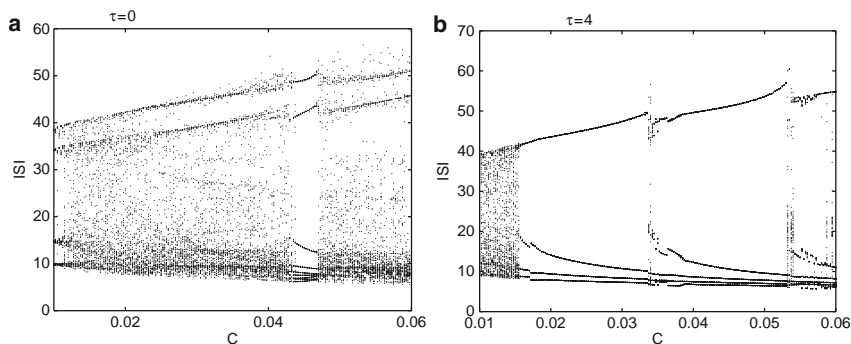
$$S(\tau_1) = \left( \frac{\langle (V_1(t) - V_2(t - \tau_1))^2 \rangle}{(\langle V_1^2(t) \rangle \langle V_2^2(t) \rangle)^{\frac{1}{2}}} \right)^{\frac{1}{2}} \quad (10.6)$$

where  $\langle \cdot \rangle$  denotes the temporal average. In the numerical simulation, this value can be calculated with enough long time after omitting the enough transient states. This function measures the temporal correlation of two signals  $V_1(t), V_2(t)$  with the property given below. If two signals are independent, then  $S(\tau_1) \neq 0$  for any  $\tau_1$ . If  $S(\tau_1) \approx 0$  for certain  $\tau_1$ , then there exists a time lag  $\tau_1$  between two signals. Hence for complete synchronization states, the similarity function  $S(\tau_1)$  has a minimum zero at  $\tau_1 = 0$ . Now we consider the case of complete synchronization of (10.5) here and calculate  $S(0)$  when the time delay  $\tau$  and the coupling strength  $C$  vary. A contour graph is plotted in  $(C, \tau)$ -parameter plane as shown in Fig. 10.4a. One can see clearly in Fig. 10.4b that there exist extensive regions where  $S(0)$  vanishes, and this implies the occurrence of complete synchronization of two coupled ML neurons.

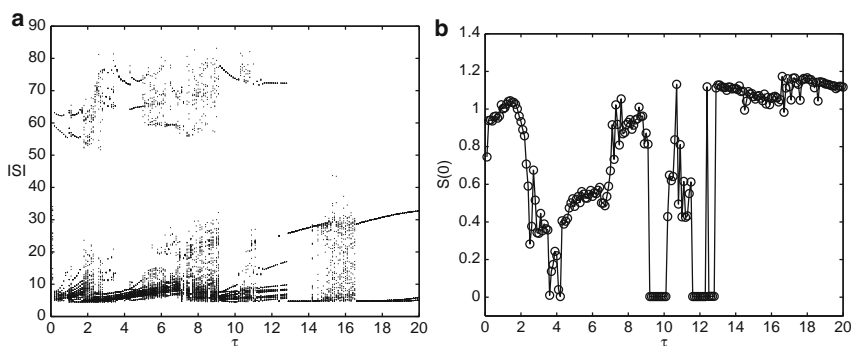
However, it is worth noting that when two chaotic-coupled neurons achieve complete synchronization in the presence of time delay they exhibit regular firing patterns



**Fig. 10.4** **a** The contour plot of  $S(0)$  in  $(\tau, C)$ -parameter plane, where the gray scale table on the bar shows its variation; **b**  $S(0)$  vs. the coupling strength  $C$  for the delay  $\tau = 4$



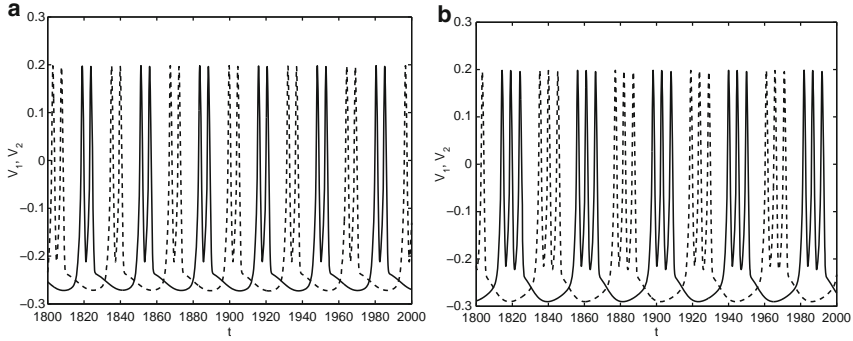
**Fig. 10.5** **a** The bifurcation diagram of ISI with respect to the coupling strength  $C$  of the first neuron without time delay; **b** the bifurcation diagram of ISI of the first neuron with respect to the coupling strength  $C$  for the delay  $\tau = 4$



**Fig. 10.6** **a** The bifurcation diagram of ISI of the first neuron with respect to the delay  $\tau$ ; **b**  $S(0)$  vs. the delay  $\tau$  for the coupling strength  $C = 0.12$

instead of original chaotic behavior (see Fig. 10.5). Although the neurons can exhibit rich firing patterns as the delay is changed, complete synchronization can only occur in periodic behavior.

For further investigation, we can find that delay can destroy synchronization or induce antiphase synchronization for certain delays. When we take the coupling strength  $C = 0.12$ , where the coupled neurons can achieve synchronization without delay, the bifurcation diagram of ISI and the similarity function  $S(0)$  with respect to the delay are shown in Fig. 10.6a, b respectively. It is obvious that the synchronization of coupled neurons is destroyed for large ranges of delay and different firing patterns may occur. Moreover, for some delays, the coupled neurons can realize antiphase synchronization. One can see that antiphase synchronization with periodic firing patterns can appear in Figs. 10.7a, b for the delay  $\tau = 14$  and 13, respectively.



**Fig. 10.7** **a** Antiphase synchronization with period two for the delay  $\tau = 14$ ; **b** Antiphase synchronization with period three for the delay  $\tau = 13$  with the coupling strength  $C = 0.12$

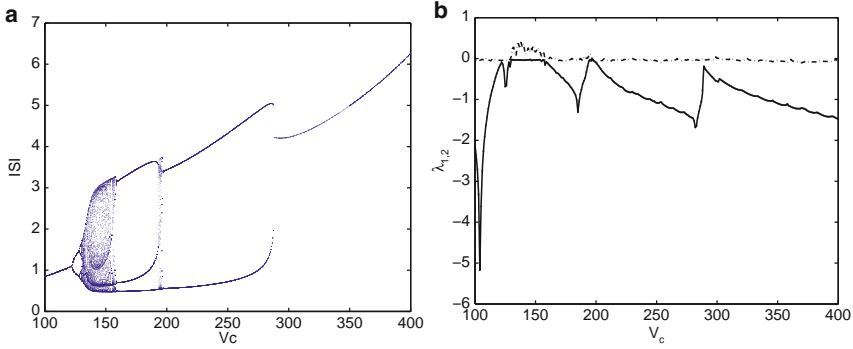
## 10.4 Synchronization and Firing Patterns in Coupled Neurons with Delayed Inhibitory Synapses

In [23], It was shown that with the control parameter  $V_c$  changing, the Chay neuron model (see Appendix 10.3) exhibits rich firing behavior, such as periodic spiking and bursting and chaotic patterns. The bifurcation diagram of ISI and the corresponding two largest Lyapunov exponents with respect to the parameter  $V_c$  for a single Chay neuron model are shown in Fig. 10.8.

The dynamics of two coupled Chay neurons with delayed inhibitory synapses are governed by the following set of differential equations:

$$\begin{aligned} \frac{dV_{1,2}}{dt} &= g_I m_\infty^3 h_\infty (V_I - V_{1,2}) + g_{Kv} n^4 (V_K - V_{1,2}) + \frac{g_{Kc} C_{1,2}}{1 + C_{1,2}} (V_K - V_{1,2}) \\ &\quad + g_I (V_I - V_{1,2}) + \frac{H_{\text{syn}} (V_{\text{syn}} - V_{1,2})}{1 + \exp(-\sigma(V_{2,1}(t - \tau) - \theta))}, \\ \frac{dn_{1,2}}{dt} &= \frac{n_\infty - n_{1,2}}{\tau_n}, \\ \frac{dC_{1,2}}{dt} &= \rho [m_\infty^3 h_\infty (V_c - V_{1,2}) - K_c C_{1,2}], \end{aligned}$$

where  $H_{\text{syn}}$  is the coupling strength and  $\tau$  is the time delay in two coupled Chay neurons with synaptic connection. The subscript 1 (or 2) represents the neuron 1 (or 2).  $V_{\text{syn}}$  is the synaptic reversal potential, which is dependent on the type of synaptic transmitter released from a presynaptic neuron and its receptors. The coupling becomes excitatory or inhibitory with  $V_{\text{syn}} > V_e$  or  $V_{\text{syn}} < V_e$ , where  $V_e$  is an equilibrium potential for a neuron.  $\theta$  is the synaptic threshold, above which the postsynaptic neuron is affected by the presynaptic one;  $\sigma$  represents a constant rate of onset of excitation or inhibition.



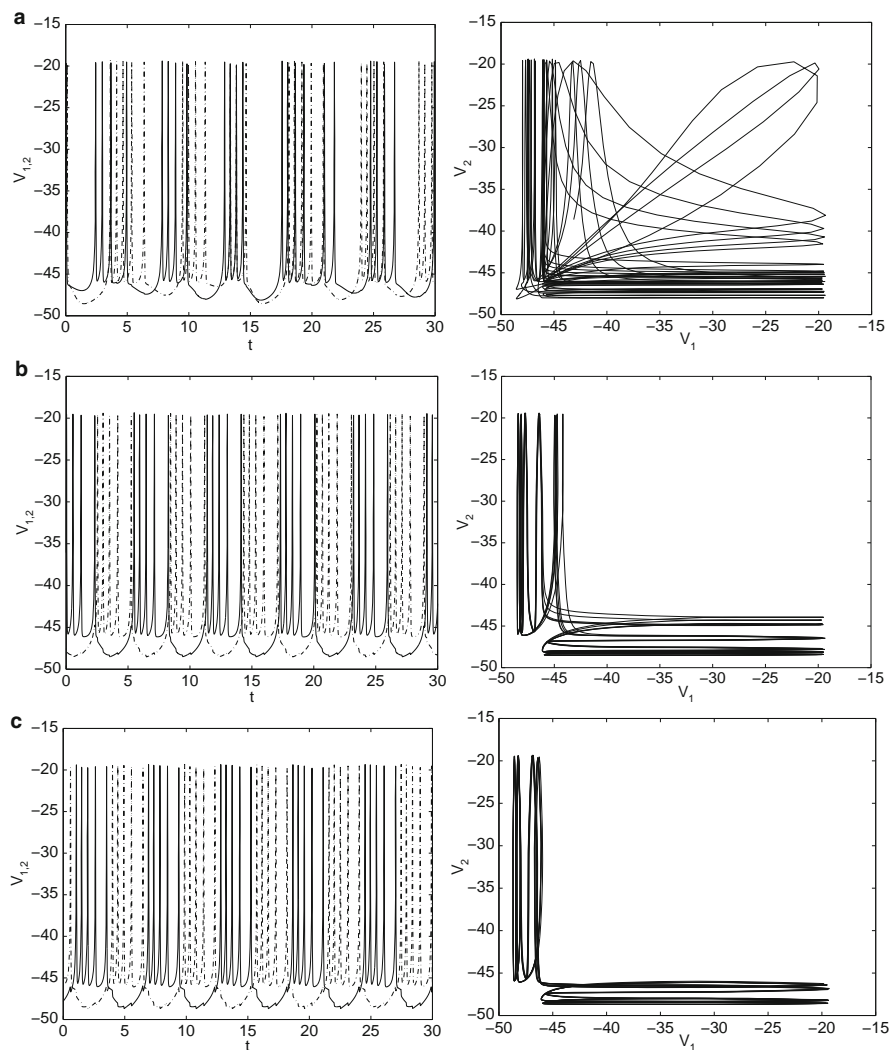
**Fig. 10.8** **a** The bifurcation diagram of ISI vs. the parameter  $V_c$  in the Chay neuron model; **b** the corresponding two largest Lyapunov exponents  $\lambda_{1,2}$  of **a**

Here, we choose the synaptic reversal potential  $V_{\text{syn}} = -50$  mV so that interaction of two neurons is inhibitory, with the synaptic threshold  $\theta = -30$  and the rate constant  $\sigma = 10$ .

To begin with, we take a look at the synchronization of two synaptically coupled chaotic Chay neurons without conduction delay (i.e.,  $\tau = 0$ ). When we choose the control parameter  $V_c = 136$  mV, a single Chay neuron exhibits chaotic firing behavior as shown in Fig. 10.8. With the coupling strength  $H_{\text{syn}}$  increasing from initially zero, two coupled chaotic Chay neurons gradually transit from nonsynchronization to out-of-phase bursting synchronization. Meanwhile, two chaotic neurons become more and more regular in the process of realizing out-of-phase synchronization. This is resulted from the effect of inhibition, which forces one neuron to be active and other one to be quiescent alternately under a sufficient strong coupling. In order to observe the transition to out-of-phase synchronization clearly with the coupling strength  $H_{\text{syn}}$  increasing, we take the coupling strength  $H_{\text{syn}} = 1, 3.5$ , and  $6$ , respectively. Temporal evolutions of the corresponding membrane potential and the phase portraits on  $(V_1, V_2)$ -plane of two synaptically coupled chaotic Chay neurons are shown in Fig. 10.9. It is obvious that out-of-phase bursting synchronization with a period-5 firing can be achieved when the coupling strength  $H_{\text{syn}}$  is above a certain critical value.

In what follows, we focus on the effect of the conduction delay on the synchronization in two coupled Chay neurons. Three cases are considered in terms of the states of coupled neurons. Typically, when we take the coupling strength  $H_{\text{syn}} = 1, 3.5$ , and  $6$ , two coupled neurons lie in the states of nonsynchronization, nearly out-of-phase synchronization and out-of-phase synchronization, respectively. The similarity function  $S_V(0)$  is also introduced as that in Sect. 10.3 to investigate the effect of conduction delay.

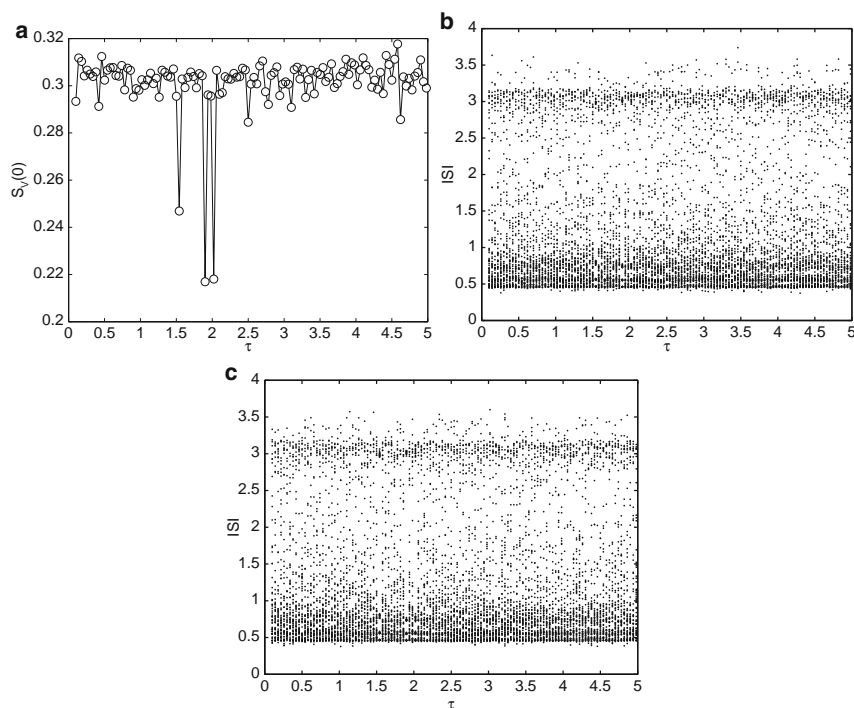
At first, it is shown that the conduction delay cannot efficiently enhance the synchronization of two coupled Chay neurons when two coupled Chay neurons are in nonsynchronous state. From the results of numerical simulation for  $H_{\text{syn}} = 1$



**Fig. 10.9** The time series of two coupled Chay neurons, denoted by *solid* and *dashed-dot* curves, respectively, and the corresponding phase portraits on  $(V_1, V_2)$  plane, for the coupling strength **a**  $H_{\text{syn}} = 1$ ; **b**  $H_{\text{syn}} = 3.5$ ; **c**  $H_{\text{syn}} = 6$

illustrated in Fig. 10.10, it is obvious that the similarity function  $S_V(0)$  cannot be reduced drastically by the conduction delay and two coupled neurons are not synchronous and still in chaotic firing within the range of conduction delay concerned. There are only several isolated points, at which  $S_V(0)$  is lightly minimized with a little enhancement of synchronization in two coupled neurons.

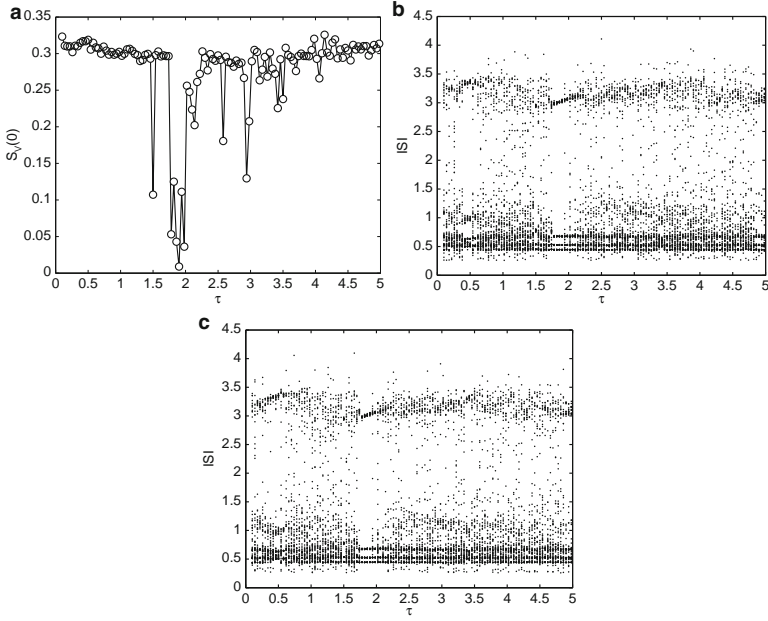
Next, if two coupled neurons lie in nearly out-of-phase synchronization state for  $H_{\text{syn}} = 3.5$  as shown in Fig. 10.9b, what influence does the conduction delay have



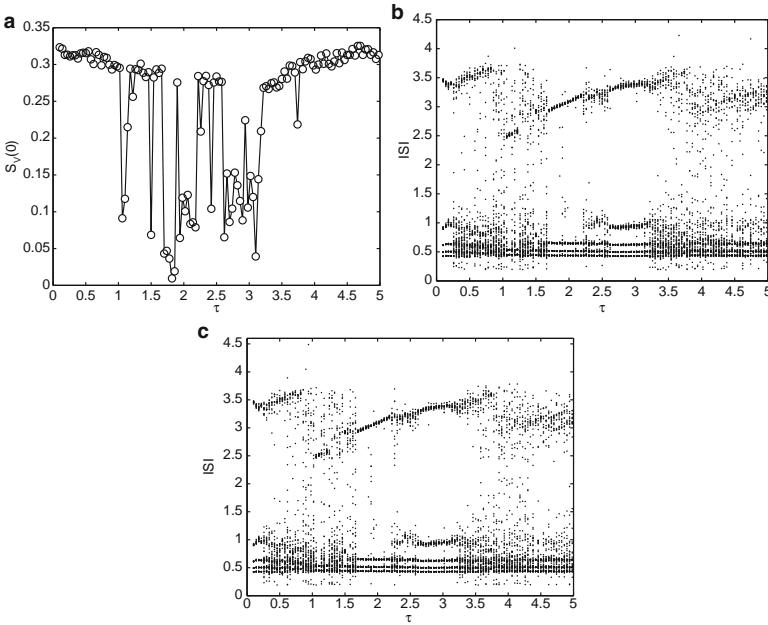
**Fig. 10.10** **a** The similarity function  $S_V(0)$  vs. the conduction delay  $\tau$ ; **b** the bifurcation diagram of ISI of the first neuron vs. the conduction delay  $\tau$ ; **c** the bifurcation diagram of ISI of the second neuron vs. the conduction delay  $\tau$ , where the coupling strength  $H_{\text{syn}} = 1$

on the synchronization of neurons? The bifurcation diagrams of ISI of two neurons and the similarity function  $S_V(0)$  with respect to the conduction delay  $\tau$  are shown in Fig. 10.11. There is a clear window, in which the similarity function  $S_V(0)$  is reduced drastically and at the same time two coupled Chay neurons behave as periodic firings in this window. Hence, the conduction delay can enhance synchronization when the coupled Chay neurons are nearly in out-of-phase synchronization. However, it is noted that the effective range of delay for enhancing synchronization is very narrow.

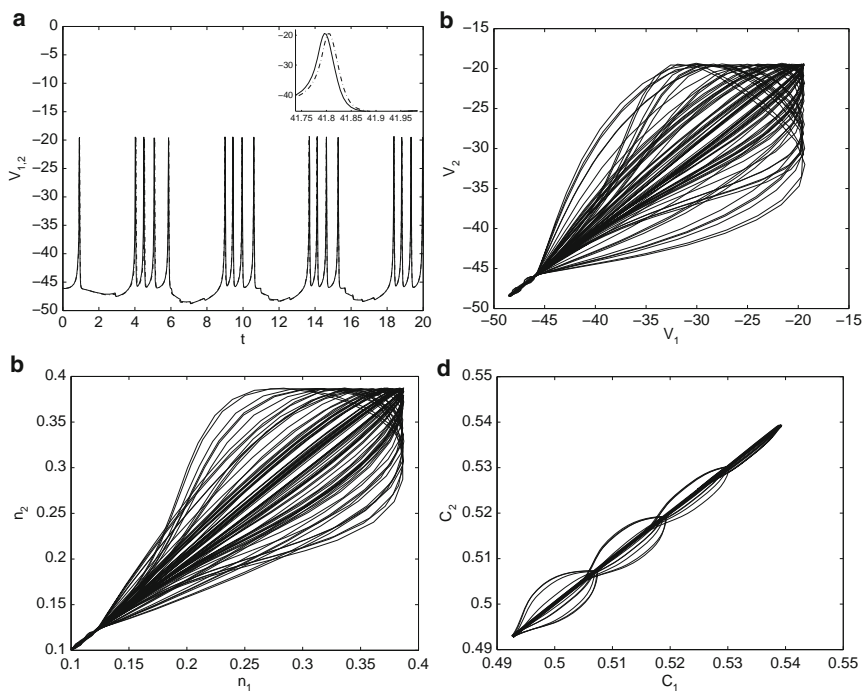
Finally, if two coupled Chay neurons can achieve out-of-phase synchronization without conduction delay for  $H_{\text{syn}} = 6$  as shown in Fig. 10.9c, it is shown that the firing behavior can be greatly changed by the conduction delay. The bifurcation diagrams of ISI and the corresponding similarity function  $S_V(0)$  vs. the conduction delay are illustrated in Fig. 10.12. It is shown that the similarity function  $S_V(0)$  can be drastically reduced for more values of conduction delay. This implies the enhancement of in-phase synchronization of two synaptically coupled neurons due to conduction delay. Moreover, we can see in Fig. 10.12 that the firings of two coupled



**Fig. 10.11** **a** The similarity function  $S_V(0)$  vs. the conduction delay  $\tau$ ; **b** the bifurcation diagram of ISI of the first neuron vs. the conduction delay  $\tau$ ; **c** the bifurcation diagram of ISI of the second neuron vs. the conduction delay  $\tau$ , where the coupling strength  $H_{\text{syn}} = 3.5$



**Fig. 10.12** **a** The similarity function  $S_V(0)$  vs. the conduction delay  $\tau$ ; **b** the bifurcation diagram of ISI of the first neuron vs. the conduction delay  $\tau$ ; **c** the bifurcation diagram of ISI of the second neuron vs. the conduction delay  $\tau$ , where the coupling strength  $H_{\text{syn}} = 6$

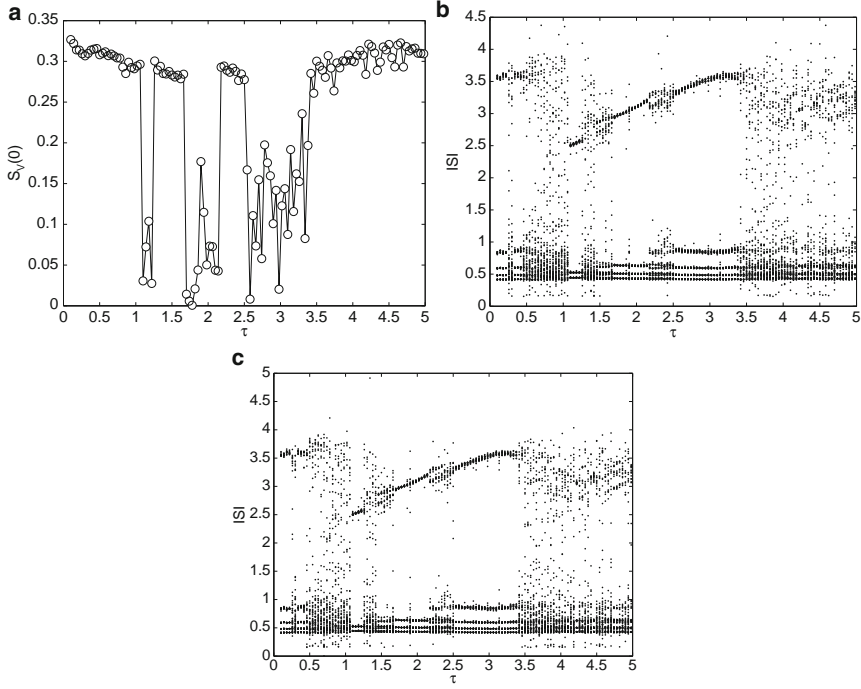


**Fig. 10.13** **a** The time series of membranes of two coupled Chay neurons; **b** the phase portraits on  $(V_1, V_2)$  plane; **c** the phase portraits on  $(n_1, n_2)$ -plane; **d** the phase portraits on  $(C_1, C_2)$ -plane; where the conduction delay  $\tau = 2$ , and the coupling strength  $H_{\text{syn}} = 6$

neurons become periodic during the enhancement of synchronization. There are two clear windows, in which synchronization and regularization of coupled neurons are greatly enhanced due to the conduction delay.

However, it is noted that enhanced synchronization is actually a bursting synchronization with spikes being weakly correlated as shown in Fig. 10.13. Furthermore, it is illustrated from Fig. 10.13b, c that the synchronization of the slow subsystems of coupled neurons is stronger than that of the fast subsystems because the correlation degree of two slow variables  $C_1$  and  $C_2$  is higher than that of the corresponding fast variables  $V_1$  and  $V_2$  or  $n_1$  and  $n_2$ .

For stronger coupling strength  $H_{\text{syn}}$ , similar investigation can be conducted (see Fig. 10.14). It is shown that with the coupling strength  $H_{\text{syn}}$  increasing, the number of windows, in which synchronization and regularization of coupled neurons are enhanced, is increased. Moreover, it is seen that synchronization cannot be enhanced at high or low conduction delays.



**Fig. 10.14** **a** The similarity function  $S_V(0)$  with respect to the conduction delay  $\tau$ ; **b** the bifurcation diagram of ISI of neuron 1 vs. the conduction delay  $\tau$ ; **c** the bifurcation diagram of ISI of neuron 2 vs. the conduction delay  $\tau$ , where the coupling strength  $H_{\text{syn}} = 8$

## 10.5 Synchronization and Firing Patterns in Coupled Neurons with Delayed Excitatory Synapses

For the fast spiking (FS) neuron model (see Appendix 10.4) with the dynamics of the voltage being as following:

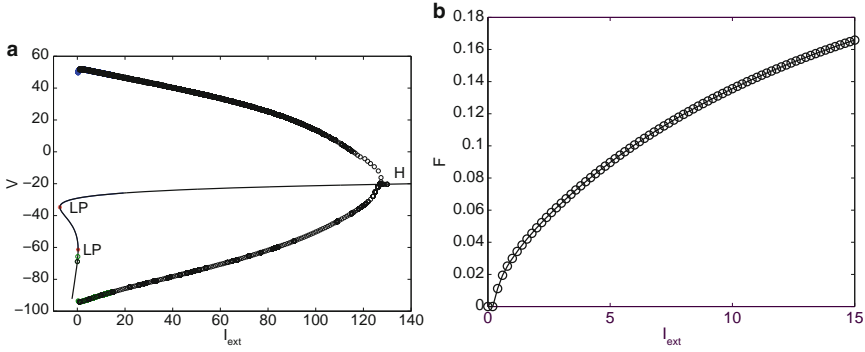
$$C \frac{dV}{dt} = g_{\text{Na}} m^3 h (V_{\text{Na}} - V) + g_{\text{Kv}} n^2 (V_{\text{K}} - V) + g_{\text{I}} (V_{\text{I}} - V) + I_{\text{ext}} + I_{\text{syn}},$$

where  $I_{\text{syn}}$  is the synaptic current resulted from all presynaptic neurons and its dynamics is described by a model of receptor binding [24]:

$$I_{\text{syn}} = -g_{\text{syn}} r (V - V_{\text{syn}}), \quad (10.7)$$

$$\frac{dr}{dt} = \alpha T (1 - r) - \beta r, \quad (10.8)$$

$$T = \frac{1}{1 + \exp(-V_{\text{pre}})}, \quad \alpha = \frac{1}{\tau_{\text{rise}}} - \beta, \quad \beta = \frac{1}{\tau_{\text{decay}}}, \quad (10.9)$$



**Fig. 10.15** **a** The bifurcation diagram of equilibrium vs. the parameter  $I_{\text{ext}}$  in the FS neuron model; **b** the corresponding  $F-I_{\text{ext}}$  curve

where  $V_{\text{syn}}$  is the synaptic reverse potential,  $V_{\text{pre}}$  is the membrane potential of the presynaptic neuron,  $\tau_{\text{rise}}$  is the rise time constant of synapse,  $\tau_{\text{decay}}$  is the decay time constant of synapse, and  $g_{\text{syn}}$  is the coupling strength. We set  $\tau_{\text{rise}} = 0.01$  ms,  $V_{\text{syn}} = 40$  mV for the excitatory chemical connections.  $I_{\text{ext}}$ ,  $\tau_{\text{decay}}$ , and  $g_{\text{syn}}$  are used as the control parameters.

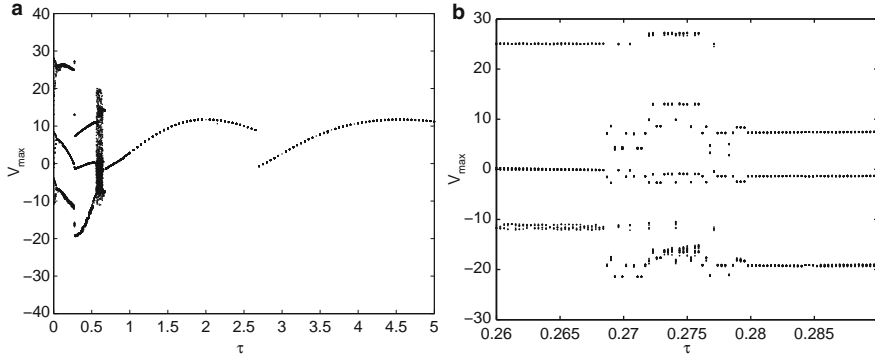
For a single FS neuron, when the parameter  $I_{\text{ext}}$  is changed, periodic firing arises at  $I_{\text{ext}} \approx 0.35$  through a saddle-node bifurcation. At  $I_{\text{ext}} = 127.5$ , the periodic firing dies out through an inverse Hopf bifurcation (see Fig. 10.15a). Hence, the FS neuron exhibits type-I excitability. The corresponding frequency vs. the external stimulus  $I_{\text{ext}}$ , that is, the  $F-I_{\text{ext}}$  curve, is shown in Fig. 10.15b, which has small frequencies and varies continuously.

In the following, the effect of the time delay on synchronization in coupled FS neurons will be studied. Suppose that the function that describes the influence of the  $i$ th unit on the  $j$ th unit at time  $t$  depends on the state of the  $i$ th unit at some earlier time  $t - \tau$ . For example, the dynamics of two coupled FS neurons with a synaptic delay is described by

$$\begin{aligned} C \frac{dV_{1,2}}{dt} &= g_{\text{Na}} m_{1,2}^3 h_{1,2} (V_{\text{Na}} - V_{1,2}) + g_{\text{Kv}} n_{1,2}^2 (V_{\text{K}} - V_{1,2}) \\ &\quad + g_{\text{l}} (V_{\text{l}} - V_{1,2}) + I_{\text{ext}} + g_{\text{syn}} r_{1,2} (t - \tau) (V_{\text{syn}} - V_{1,2}), \\ \frac{dx_{1,2}}{dt} &= \frac{x_{\infty} - x_{1,2}}{\tau_{\text{x}}(V_{1,2})}, \quad (x = m, h, n), \\ \frac{dr_{1,2}}{dt} &= \alpha T (1 - r_{1,2}) - \beta r_{1,2}, \end{aligned}$$

where  $\tau$  is the delay of information transmission through the synapse between two neurons, the subscripts 1 and 2 are for neurons 1 and 2, respectively.

To begin with, we fix the parameters  $I_{\text{ext}} = 4$ ,  $\tau_{\text{decay}} = 6.26$  and  $g_{\text{syn}} = 1.53$ . The effect of delay on the coupled neurons is shown in Fig. 10.16. It can be seen clearly that as  $\tau$  increases, the coupled neurons fire periodically or chaotically. More



**Fig. 10.16** **a** The bifurcation diagram of  $V_{\max}$  of neuron 1 vs. the delay  $\tau$  in the coupled FS neuron model; **b** an enlargement of a part of **a**

importantly, it is shown that there are two kinds of transition processes within the same periodic pattern, namely, the transition from one period-3 firing to another one for  $\tau$  between 0.26 and 0.28 (see Fig. 10.16a, b) as well as that from one period-1 firing to another one for  $\tau$  between 2.5 and 3.0 (see Fig. 10.16a), which will be discussed in what follows.

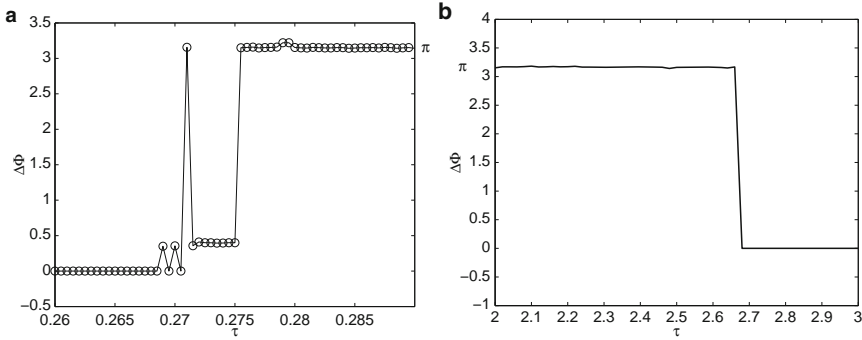
To carry out further analysis, define the phase of a neuron as follows:

$$\phi_k(t) = 2\pi \frac{t - t_n^k}{t_{n+1}^k - t_n^k} + 2n\pi, \quad t_n^k \leq t \leq t_{n+1}^k \quad (k = 1, 2). \quad (10.10)$$

where  $\{t_n^k\}$  represents the spike sequence of the  $k$ th neuron. The phase variation of two neurons can be studied through the instantaneous phase difference between them,  $\Delta\Phi(t) = |\phi_1(t) - \phi_2(t)|$ . If  $\Delta\Phi = \lim_{t \rightarrow +\infty} \Delta\Phi(t) = 0$  (or  $\pi$ ), then the coupled neurons realize in-phase (or anti-phase) synchronization.

It can be shown from Figs. 10.16 and 10.17 that there are two essentially different transition processes as follows:

1. Transition from in-phase synchronization of one period-3 firing to antiphase synchronization of another one for  $\tau$  between 0.26 and 0.28 (see Figs. 10.16b and 10.17a). This process is somewhat complicated and includes some intermediate nonsynchronous states. Here we call it as “continuous transition.” This transition process is clearly displayed in Fig. 10.18, where  $\tau = 0.265, 0.27, 0.275$ , and  $0.28$ , respectively.
2. Transition from antiphase synchronization of one period-1 firing to in-phase synchronization of another one. There is no intermediate nonsynchronous state during transition but a sudden jump between them, and then it is called “jump transition.” The corresponding numerical simulations are shown in Fig. 10.19.



**Fig. 10.17** Two cases of the variation of  $\Delta\phi$  with respect to the time delay  $\tau$ : **a** continuous transition; **b** jump transition

It should be noted that there exist even more complex transitions in other ranges of  $\tau$ , including chaotic and periodic firing modes between the antiphase synchronous states of period-3 and period-1 firings.

For other combinations of parameters, similar investigation can also be conducted and it is found that the above-derived transition modes are universal in two synaptically coupled FS neurons with time delay. For example, taking  $I_{\text{ext}} = 4$ ,  $\tau_{\text{decay}} = 5.5$ , and  $g_{\text{syn}} = 1.53$ , the corresponding numerical simulations are shown in Fig. 10.20, which are similar to Fig. 10.16. Moreover, it is shown in Fig. 10.20b that chaotic firing appears in a narrow range  $0.50 < \tau < 0.51$  between two antiphase synchronous period-1 firings.

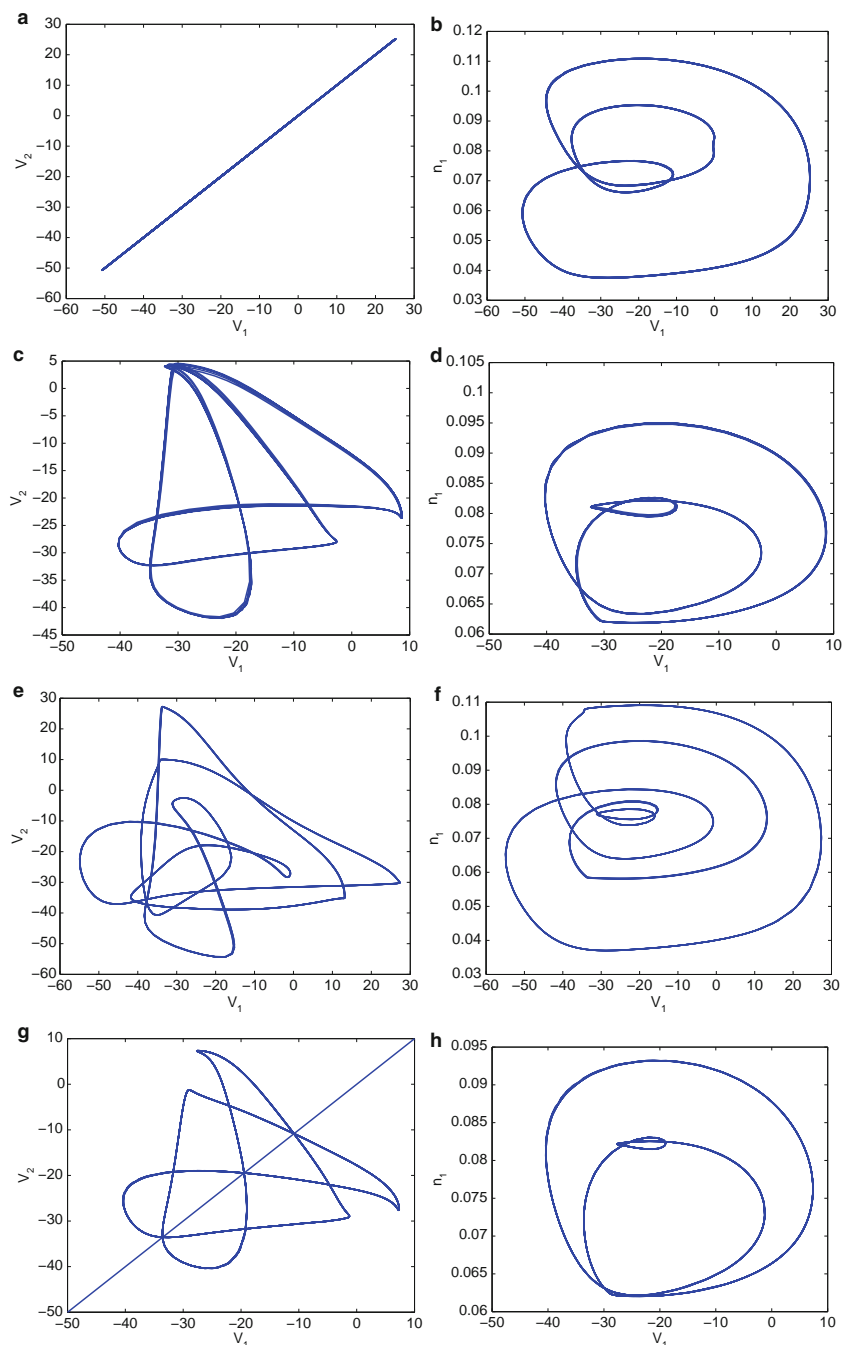
## 10.6 Delay Effect on Multistability and Spatiotemporal Dynamics of Coupled Neuronal Activity [14]

The relationship between spatial profiles of neural interactions and spatiotemporal patterns of neuronal activity can be investigated in modeling studies. It was shown that delays might lead to homogeneous oscillations in inhibitory networks with homogeneous, random connectivity. It was also shown that when interactions were spatially structured, delays induced a wealth of dynamical states with different spatiotemporal properties and domains of multistability in [14].

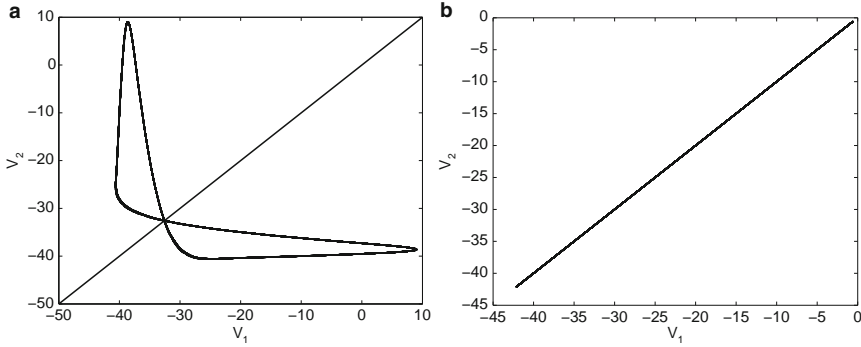
In [14], the authors considered the following equation

$$\alpha \dot{m} = -m + \Phi[I(x) + \int dy J(|x - y|) m(y, t - \tau)], \quad (10.11)$$

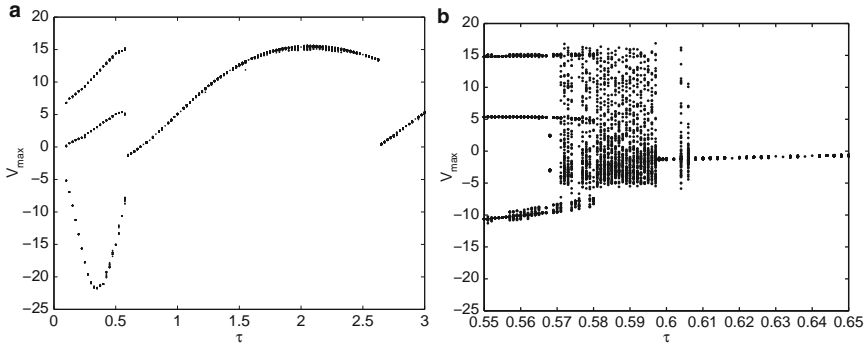
where  $m(x)$  is the activity of neurons at a one-dimensional location  $x$  on a periodic ring  $[-\pi, \pi]$ ,  $\alpha$  is the time constant of rate dynamics ( $\alpha = 1$  in the following), and  $\Phi$  is the steady-state current-to-rate transfer function with  $\Phi(I) = I$  if  $I > 0$  and  $\Phi(I) = 0$  otherwise. The total synaptic input is split into the external current  $I(x)$



**Fig. 10.18** The transition process from in-phase synchronization to antiphase synchronization in period-3 firings. The phase plots in  $(V_1, V_2)$ -plane with the attractors of neuron 1 are: **a** and **b** in-phase synchronization for  $\tau = 0.265$ ; **c** and **d** intermediate nonsynchronous state for  $\tau = 0.27$ ; **e** and **f** intermediate nonsynchronous state for  $\tau = 0.275$ ; **g** and **h** antiphase synchronization for  $\tau = 0.28$



**Fig. 10.19** The transition process from antiphase synchronization to in-phase synchronization in period-1 firings. The phase plots in  $(V_1, V_2)$ -plane are: **a** antiphase synchronization for  $\tau = 2.65$ ; **b** in-phase synchronization for  $\tau = 2.7$



**Fig. 10.20** **a** The bifurcation diagram of  $V_{\max}$  of neuron 1 vs. the delay  $\tau$  in the coupled FS neuron model; **b** an enlargement of a part of **a**

and the synaptic current due to the presynaptic activity at a location  $y$  with a weight  $J(|x - y|)$  and a delay  $\tau$ .

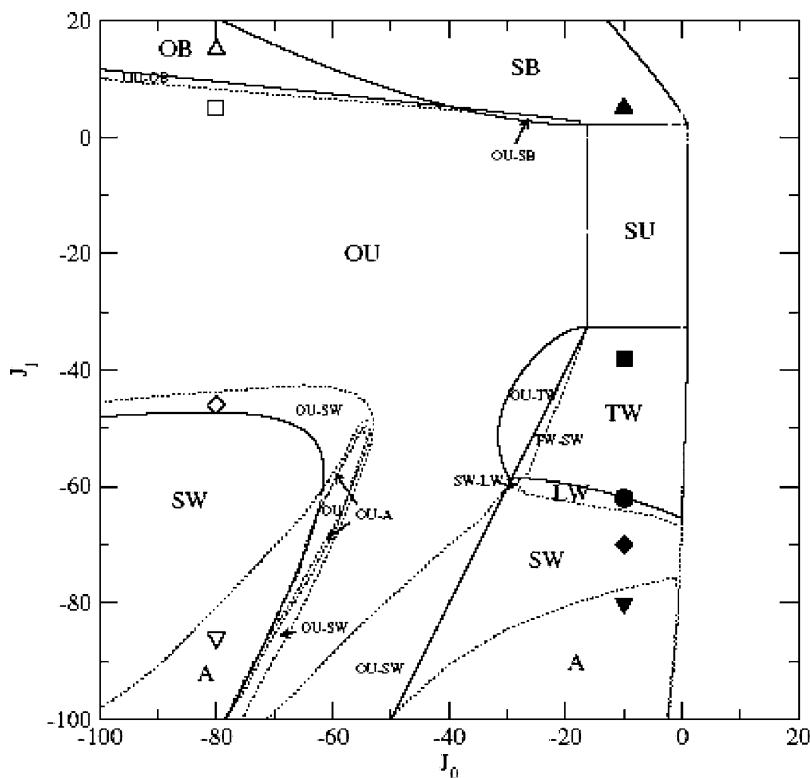
In the absence of delay (i.e.,  $\tau = 0$ ), the dynamics of (10.11) for a stationary, homogeneous external input converges to a stable fixed point, for which the activity of neuron is either homogeneous or localized, depending on the spatial modulation of the interactions. For the case of  $\tau > 0$ , the stability of the stationary uniform state with respect to perturbations of wave number  $n$  is given by the dispersion relation

$$\lambda = -1 + J_n e^{-\lambda \tau},$$

where

$$J_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} dy J(y) \times \cos ny.$$

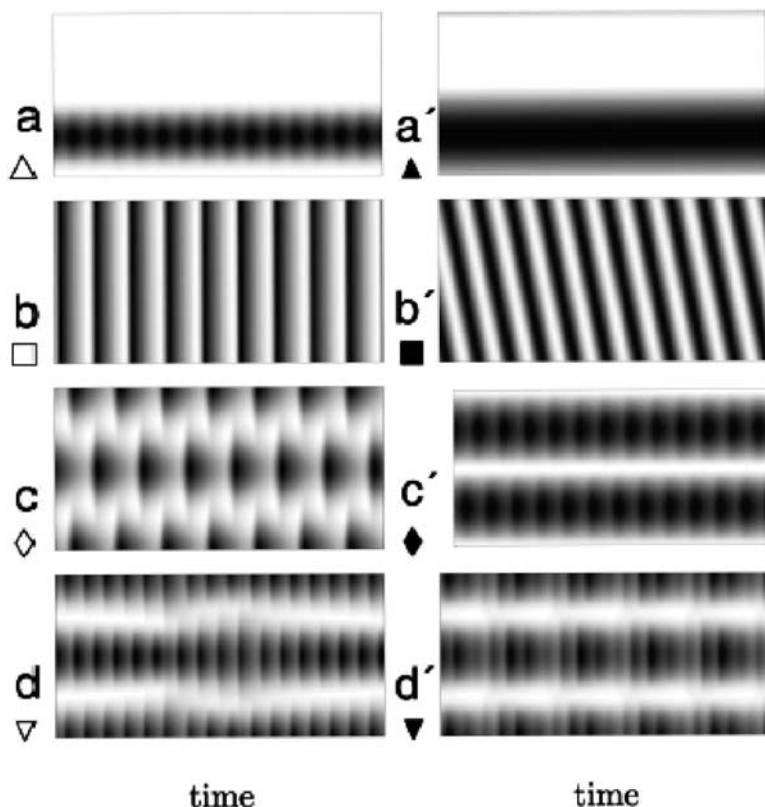
A steady instability of the  $n$ th mode occurs for  $J_n = 1$ , while for  $J_n \cos(\tau \omega) = 1$  there is an oscillatory instability with frequency  $\omega = -\tan(\tau \omega)$ . Hence, four types



**Fig. 10.21** (Cited from [14]) Phase diagram of the rate model (10.11) for  $\tau = 0.1$ . The states are referred to the patterns in Fig. 10.22. Regions of bistability are indicated by *hyphens*, e.g., OU-SW. Reprinted with permission from [14]. Copyright (2005) by the American Physical Society

of linear instability of the stationary uniform state are possible, including (1) a firing-rate instability ( $\omega = 0, n = 0$ ), (2) a Hopf instability ( $\omega \neq 0, n = 0$ ), (3) a Turing instability ( $\omega = 0, n \neq 0$ ), and (4) a Turing–Hopf instability ( $\omega \neq 0, n \neq 0$ ). In what follows, the characteristics and stability of nonlinear firing patterns arising from these instabilities are studied. For simplicity, assume that the interaction has only two nonzero Fourier components:  $J(|x - y|) = J_0 + J_1 \cos(x - y)$ . For  $J_0 > |J_1|$ , the interaction is purely excitatory, while for  $J_0 < -|J_1|$  it is purely inhibitory. For  $J_1 > |J_0|$ , the connectivity is locally excitatory and inhibitory at larger distances, while for  $J_1 < -|J_0|$  the inverse is true.

Analytical and numerical investigation of (10.11) reveals a phase diagram on  $(J_0, J_1)$ -plane shown in Fig. 10.21, in which one can discern eight states of activity: stationary uniform (SU), stationary bump (SB), oscillatory bump (OB), oscillatory uniform (OU), traveling waves (TW), standing waves (SW), lurching waves (LW), and aperiodic dynamics (A). Figure 10.22 gives the space–time plots of typical patterns of activity in the different regions shown in Fig. 10.21.



**Fig. 10.22** (Cited from [14]) Space–time plots of typical patterns of activity in the different regions of Fig. 10.21 shown over five units of time. In the *left-hand column* from *top to bottom*,  $J_0 = -80$  and  $J_1 = 15, 5, -46, -86$ , respectively, correspond to the states OB, OU, SW, and A in Fig. 10.21. In the *right-hand column* from *top to bottom*,  $J_0 = -10$  and  $J_1 = 5, -38, -70, -80$ , respectively, correspond to the states SB, TW, SW, and A.  $\tau = 0.1$  and  $I$  is varied to maintain the mean firing rate at 0.1. Dark regions indicate higher levels of activity in gray scale. Reprinted with permission from [14]. Copyright (2005) by the American Physical Society

The results presented are for a threshold-linear transfer function and simplified connectivity. While the simplicity of (10.11) allows for analysis, firing-rate models do not necessarily provide an accurate description of the dynamics of more realistic networks of spiking neurons (NSN). To what extent are the dynamics in (10.11) relevant for understanding the patterns of activity observed in the NSN? Now consider a one-dimensional network of conductance-based neurons with periodic boundary conditions, composed of two populations of  $n$  neurons: excitatory  $E$  and inhibitory  $I$ . All neurons are described by a Hodgkin–Huxley type model (see Appendix 10.1) with one somatic compartment, in which Na and K ionic currents shape the action potentials. The probability of connection from a neuron in the population  $A \in (E, I)$

to a neuron in another population  $B$  is  $p_{BA}$ , where  $p$  depends on the distance  $r$  between them as  $p_{BA} = p_0^{BA} + p_1^{BA} \cos(r)$ . Synaptic currents are modeled as

$$I_{\text{syn},A} = -g_A s(t)(V - V_A), \quad A \in (E, I),$$

where  $V$  is the voltage of the postsynaptic neuron,  $V_A$  is the reversal potential of the synapse (0 mV for  $A = E$  and  $-880$  mV for  $A = I$ ),  $g_A$  is the maximum conductance change, and  $s(t)$  is a variable which, given a presynaptic spike at time  $t^* - \delta$ , takes the form

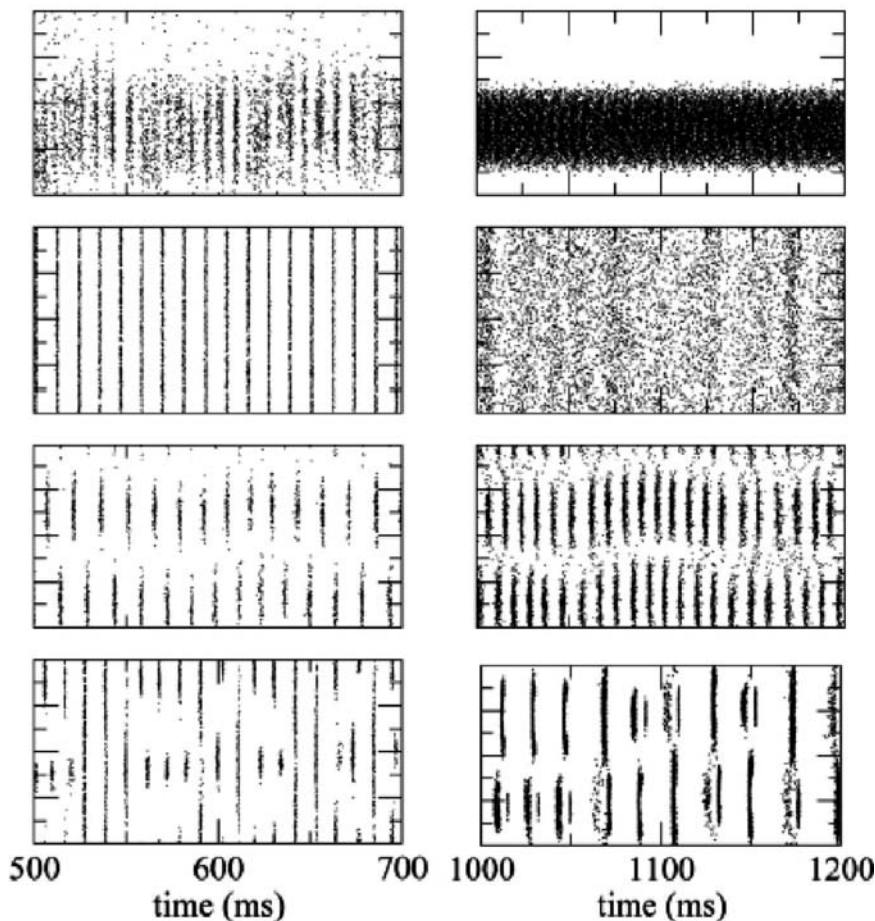
$$s(t) = \frac{1}{\tau_1 - \tau_2} (e^{-(t-t^*)/\tau_1} - e^{-(t-t^*)/\tau_2}),$$

where  $\delta$  is the delay and  $\tau_1$  and  $\tau_2$  are the rise and decay times. Each neuron receives an external, excitatory synaptic input as a Poisson process with a rate  $v_{\text{ext}}$ , modeled as the synaptic currents above with a maximum conductance change of  $g_{\text{ext}}$ . Choose probabilities  $p_{IE} = p_{EE} = p_E$ ,  $p_{EI} = p_{II} = p_I$  and identical synaptic time constants for excitatory and inhibitory connections  $\tau_1 = 1$ , ms and  $\tau_2 = 3$ , ms. This creates an effective one-population network with an effective coupling similar to that of the rate model. Figure 10.23 shows eight typical firing patterns in the NSN. The figures have been arranged to allow comparison with those in Fig. 10.22. From the eight regions of bistability displayed by (10.11), at least one (OU-SW) is also presented in the NSN in an analogous parameter regime (see Fig. 10.24).

## 10.7 Closure

In modeling realistic neuronal networks, it is important to consider time delays explicitly in the description of the transfer of information in neuronal networks. This enables us to model the complicated processes that take place in real synapses by interaction terms with an explicit time-lag in the dynamical equations of coupled neurons. It is known that the values of time delay in a realistic domain can change the qualitative properties of dynamics, such as introduce or destroy stable oscillations, enhance or suppress synchronization between different neurons, and generate spatiotemporal patterns. Some recent results on the complex dynamical behavior of delayed coupled neurons in this respect are presented in this chapter. Most of the phenomena are illustrated by numerical simulations; however, the physiological mechanisms of such stimulations remain unclear and then further qualitative analyses on them are necessary.

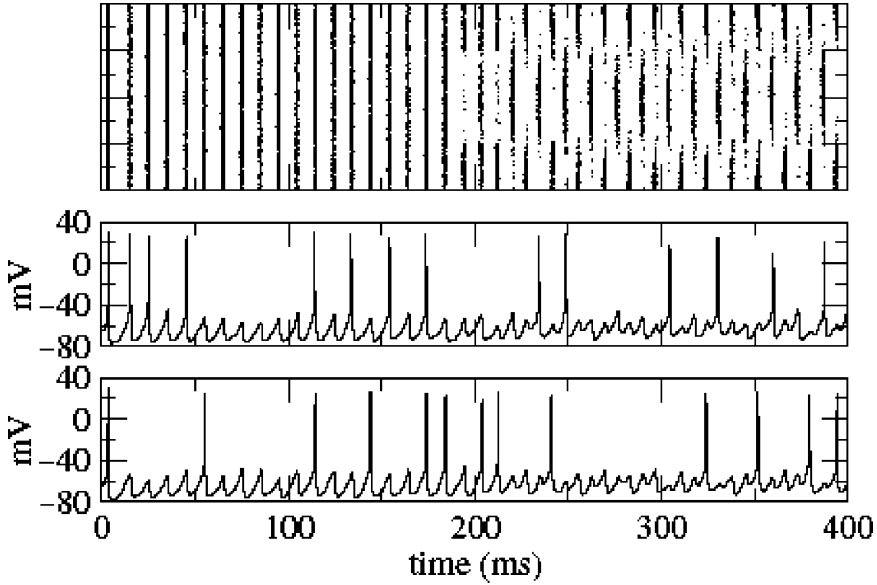
Although the importance of effects of delay in neural firing patterns has been recognized, there is an increasing interest in more complex behavior of time delays. For example, it is attractive to study spatiotemporal behavior of neural networks with a larger number of different delayed coupled neurons, possibly with an additive or multiplicative noise and complex topological structures. The results derived in that framework allow us to understand the origin of the diversity of dynamical states observed in large neural networks of spiking neurons. The phenomenon of



**Fig. 10.23** (Cited from [14]) Typical firing patterns in a NSN. The network consists of two populations of 2,000 neurons. In the *left-hand column* from *top to bottom*: a localized bump with oscillations, homogeneous oscillations, a period-doubled state of oscillating bumps, and a chaotic state. In the *right-hand column* from *top to bottom*: a steady and localized bump, the stationary uniform state, oscillating bumps and a chaotic state. Reprinted with permission from [14]. Copyright (2005) by the American Physical Society

enhanced neural synchrony by delay has important implications, in particular, in understanding synchronization of distant neurons and information processing in the brain. Moreover, synchronization of neurons was reported to play the crucial role in the emergence of pathological rhythmic brain activity in Parkinson's disease, essential tremor, and epilepsies. Obviously, the technique for suppression of undesired neural synchrony by means of delay will be an important clinical topic.

Therefore, in spite of successful experimental and clinical studies followed by biological physical experiments and medical applications, the development of effective theoretical and stimulation techniques in the studies of synchronization and firing



**Fig. 10.24** (Cited from [14]) Bistability between uniform oscillations and an oscillating bump state in a purely inhibitory network. A 30-ms inhibitory pulse is applied to neurons 500–1,500 around time 200. This induces a state in which two groups of neurons oscillate out of phase with one another; cf. voltage trace of neuron 1,000 (*middle*) and neuron 1 (*bottom*). Reprinted with permission from [14]. Copyright (2005) by the American Physical Society

patterns in coupled neuronal systems with time delay is a challenging problem of computing neuroscience and biological mathematics.

**Acknowledgments** This work was supported by the National Natural Science Foundation of China (Nos. 10432010 and 10872014) and China Post-doctoral Science Foundation (No. 20070410022). The authors also thank for the permission of reprinting from Roxin A, Brunel N, Hansel D, *Phys. Rev. Lett.* 90: 238103, 2005.

## Appendix 10.1 The Hodgkin–Huxley (HH) Model

Hodgkin and Huxley performed experiments on the giant axon of the squid and found three different types of ionic current: sodium, potassium, and leak currents. Specific voltage-dependent ion channels, one for sodium and another one for potassium, control the flow of those ions through the cell membrane. The HH neuron model [25] is described by the following differential equations:

$$C \frac{dV}{dt} = g_{\text{Na}} m^3 h (V_{\text{Na}} - V) + g_{\text{K}} n^4 (V_{\text{K}} - V) + g_{\text{L}} (V_{\text{L}} - V) + I_{\text{ext}}$$

$$\begin{aligned}\frac{dm}{dt} &= \alpha_m(V)(1-m) - \beta_m(V)m, \\ \frac{dh}{dt} &= \alpha_h(V)(1-h) - \beta_h(V)h, \\ \frac{dn}{dt} &= \alpha_n(V)(1-n) - \beta_n(V)n,\end{aligned}$$

where  $C$  is the membrane capacitance,  $V$  is the membrane potential, and  $I_{\text{ext}}$  is an externally applied current. And  $V_{\text{Na}}$ ,  $V_{\text{K}}$ , and  $V_{\text{L}}$  are the reversal potentials for  $\text{Na}^+$ ,  $\text{K}^+$ , and the leakage ions, respectively.  $g_{\text{Na}}$ ,  $g_{\text{K}}$ , and  $g_{\text{L}}$  are the maximal conductances.  $m$  and  $h$  are the activation and inactivation variables of sodium channel, and  $n$  is the activation variable of potassium channel, with

$$\begin{aligned}\alpha_n &= 0.01(V+10)/(e^{0.1V+1}-1), \\ \beta_n &= 0.125e^{V/80}, \\ \alpha_m &= 0.1(25+V)/(e^{0.1V+2.5}-1), \\ \beta_m &= 4e^{V/18}, \\ \alpha_h &= 0.07e^{V/20}, \\ \beta_h &= 1/(e^{0.1V+3}-1).\end{aligned}$$

## Appendix 10.2 The Morris–Lecar(ML) Model

The ML neuron model [26], which is a model for electrical activity in the barnacle muscle fiber, is a simplified version of the HH neuron model for describing the firing and the refractory properties of real neurons. Dynamics of the reduced ML neuron is described by the following differential equations:

$$\begin{aligned}\frac{dV}{dt} &= g_{\text{Ca}}m_{\infty}(V)(V_{\text{Ca}}-V) + g_{\text{K}}\omega(V_{\text{K}}-V), \\ &\quad + g_{\text{L}}(V_{\text{L}}-V) - I \\ \frac{d\omega}{dt} &= \lambda_{\infty}(V)(\omega_{\infty}(V) - \omega)\end{aligned}$$

with

$$\begin{aligned}m_{\infty}(V) &= 0.5 \left( 1 + \tanh \frac{V - V_{\text{a}}}{V_{\text{b}}} \right), \\ \omega_{\infty}(V) &= 0.5 \left( 1 + \tanh \frac{V - V_{\text{c}}}{V_{\text{d}}} \right), \\ \lambda_{\infty}(V) &= \frac{1}{3} \cosh \frac{V - V_{\text{c}}}{2V_{\text{d}}},\end{aligned}$$

where  $t$  is the time variable,  $V$  is the membrane action potential of the ML neuron model, and  $\omega$  is the probability of potassium ( $K^+$ ) channel activation. The values of the parameters and their detailed explanation are given in [26].

### Appendix 10.3 The Chay Model

Based on the HH model, the Chay neuron model [27] contains ion channel dynamics of HH type (sodium ( $Na^+$ ) and potassium ( $K^+$ ) as well as that of calcium ( $Ca^{2+}$ )). It was shown that the Chay model could simulate many discharging patterns of  $\beta$  cells, neuronal pacemakers, and cold receptors. The Chay model is described by the following differential equations of three variables:

$$\begin{aligned}\frac{dV}{dt} &= g_I m_\infty^3 h_\infty (V_I - V) + g_{kv} n^4 (V_k - V) + \frac{g_{kc} C}{1 + C} (V_k - V) + g_l (V_l - V), \\ \frac{dn}{dt} &= \frac{n_\infty - n}{\tau_n}, \\ \frac{dC}{dt} &= \rho [m_\infty^3 h_\infty (V_c - V) - K_c C],\end{aligned}$$

where  $t$  is the time variable,  $V$  represents the membrane action potential of the Chay neuron,  $n$  is the probability of potassium ( $K^+$ ) channel activation, and  $C$  is the dimensionless intracellular concentration of calcium ion ( $Ca^{2+}$ ); and  $V_I$ ,  $V_k$ ,  $V_c$  and  $V_l$  are the reversal potentials for the mixed  $Na^+ - Ca^{2+}$ ,  $K^+$ ,  $Ca^{2+}$  channels, and the leakage ions, respectively.  $g_I$ ,  $g_{kv}$ ,  $g_{kc}$  and  $g_l$  are the maximal conductances divided by the membrane capacitance.  $\rho$  is a proportional constant.  $K_c$  is the rate constant for efflux of intracellular  $Ca^{2+}$  ion.  $\tau_n$  is the relaxation time of the voltage-gated  $K^+$  channel;  $n_\infty$  is the steady state value of  $n$ ;  $m_\infty$  and  $h_\infty$  are the probability of activation and inactivation of the mixed channel. The explicit expressions for  $m_\infty$ ,  $h_\infty$ , and  $n_\infty$  can be written as  $y_\infty = \alpha_y / (\alpha_y + \beta_y)$ , where  $y$  stands for  $m$ ,  $n$ , and  $h$ , and

$$\begin{aligned}\alpha_m &= 0.1(25 + V) / (1 - e^{-0.1V - 2.5}), \\ \beta_m &= 4e^{-(V+50)/18} \alpha_h = 0.07e^{-0.05V - 2.5}, \\ \beta_h &= 1 / (1 + e^{-0.1V - 2}), \\ \alpha_n &= 0.01(20 + V) / (1 - e^{-0.1V - 2}), \\ \beta_n &= 0.125e^{-(V+30)/80}.\end{aligned}$$

In the numerical simulations of Sect. 10.4, the parameter values are chosen as  $g_I = 1,800$ ,  $g_{kv} = 1,700$ ,  $g_{kc} = 10$ ,  $g_l = 7$ ,  $\rho = 0.27$ ,  $K_c = 3.3/18$ ,  $V_I = 100$ ,  $V_k = -75$ , and  $V_l = -40$ , and  $V_c$  is used as a control parameter.

## Appendix 10.4 The Fast-Spiking (FS) Model

Many models of neurons have been proposed in the past with the aim of characterizing their electrophysiological properties and elucidating their roles in various cortical functions. Erisir et al. [28] carried out some pharmacological experiments on the interneurons of the mouse somatosensory cortex and found that Kv3.1/3.2 voltage-gated  $K^+$  channels play significant roles in creating a characteristic feature of fast-spiking (FS) cells. To describe the characteristic feature of FS neurons, a model was proposed as follows:

$$C \frac{dV}{dt} = g_{Na} m^3 h (V_{Na} - V) + g_{Kv} n^2 (V_k - V) + g_l (V_l - V) + I_{ext} + I_{syn},$$

$$\frac{dx}{dt} = \frac{x_\infty - x}{\tau_x(V)}, \quad (x = m, h, n),$$

$$x_\infty = \frac{\alpha_x}{\alpha_x + \beta_x}, \quad \tau_x(V) = \frac{1}{\alpha_x + \beta_x},$$

$$\alpha_m = \frac{40(75 - V)}{\exp(\frac{75-V}{13.5}) - 1},$$

$$\beta_m = 1.2262 \exp\left(\frac{-V}{42.248}\right),$$

$$\alpha_h = 0.0035 \exp\left(\frac{-V}{24.186}\right),$$

$$\beta_h = \frac{0.017(-51.25 - V)}{\exp(\frac{-51.25-V}{5.2}) - 1},$$

$$\alpha_n = \frac{95 - V}{\exp(\frac{95-V}{11.8}) - 1},$$

$$\beta_n = 0.025 \exp\left(\frac{-V}{22.222}\right),$$

where  $V$  is the membrane potential,  $m$  and  $h$  are the activation and inactivation variables of the sodium channel, respectively, and  $n$  is the activation variable of the potassium channel.

In Sect. 10.5, the parameter values are chosen as  $V_{Na} = 55.0$  mV,  $V_k = -97.0$  mV,  $V_l = -70.0$  mV,  $g_{Na} = 112$  cm<sup>-2</sup>,  $g_k = 224$  ms cm<sup>-2</sup>,  $g_l = 0.1$  ms/cm<sup>2</sup>, and  $C = 1$  μF cm<sup>-2</sup>.  $I_{ext}$  is an external stimulus current.  $I_{syn}$  is synaptic current, which results from all presynaptic neurons.

## References

1. Scott A (2002) *Neuroscience – A Mathematical Premier*. Springer, New York, NY
2. Bear MF, Connors BW, Paradiso MA (2nd Edition, 2002) *Neuroscience – Exploring the Brain*. Lippincott Williams & Wilkins., Baltimore, MD
3. Borisyuk A, Friedman A, Ermentrout B, Terman D (2005) *Mathematical Neuroscience*. Springer, Berlin
4. Koch C, Laurent G (1999) *Science* 284: 96–98
5. Crock SM, Ermentrout GB, Vanier MC, Bower JM (1997) *J. Comp. Neurosci.* 4: 161–172
6. Campbell SA (2007) Time delay in neural systems. In: McIntosh R, Jirsa VK (eds) *Handbook of Brain Connectivity*. Springer, New York, NY
7. Foss J, Milton JG (2000) *J. Neurophysiol.* 84: 975–985
8. Reddy DV, Sen A, Johnston GL (2000) *Phys. Rev. Lett.* 85: 3381–3384
9. Winkler S, Rand R (2002) *Nonlinear Dynam.* 30: 205–221
10. Rossoni E, Chen YH, Ding MZ, Feng JF (2005) *Phys. Rev. E* 71: 061904
11. Dhamala M, Jirsa VK, Ding MZ (2004) *Phys. Rev. Lett* 92: 074104
12. Burić N, Ranković D (2007) *Phys. Lett. A* 363: 282–289
13. Rosenblum M, Pikovsky A (2004) *Phys. Rev. E* 70: 041904
14. Roxin A, Brunel N, Hansel D (2005) *Phys. Rev. Lett.* 90: 238103
15. Hasegawa H (2004) *Phys. Rev. E* 70: 021912
16. Sainz-Trapaga M, Masoller C, Braun HA, Huber MT (2004) *Phys. Rev. E* 70: 031904
17. Burić N, Todorović K, Vasović N (2007) *Chaos, Solitons and Fractals*, doi: 10.1016/j.chaos.2007.08.067
18. Burić N, Todorović K, Vasović N (2007) *Phys. Rev. E* 75: 067204
19. Izhikevich EM (2000) *Int. J. Bifur Chaos* 10: 1171–1266
20. Boccaletti S, Kurths J, Osipov G (2002) *Phys. Rep.* 366: 1–101
21. Hoppensteadt FC and Izhikevich EM (1997) *Weakly Connected Neural Networks*, Springer, New York NY
22. Wang QY, Lu QS (2005) *Chinese Phys. Lett.* 22: 543–546
23. Wang QY, Lu QS, Zheng YH (2005) *Acta Biophys. Sinica* 21: 449–455
24. Destexhe A, Mainen ZF, Sejnowski. TJ. (1994) *Neural Comput.* 6: 14–18
25. Hodgkin AL, Huxley AF (1952.) *J. Physiol* 117: 500–544
26. Morris C, Lecar H (1981) *Biophys. J.* 35: 193–213
27. Chay TR (1985) *Phys. D* 16: 233–242
28. Erisir A, Lau D, Rudy B, Leonard CS (1999) *J. Neurophysiol.* 82: 2476–2489

## Chapter 11

# Delayed Random Walks: Investigating the Interplay Between Delay and Noise

Toru Ohira and John Milton

**Abstract** A model for a 1-dimensional delayed random walk is developed by generalizing the Ehrenfest model of a discrete random walk evolving on a quadratic, or harmonic, potential to the case of non-zero delay. The Fokker–Planck equation derived from this delayed random walk (DRW) is identical to that obtained starting from the delayed Langevin equation, i.e. a first-order stochastic delay differential equation (SDDE). Thus this DRW and SDDE provide alternate, but complimentary ways for describing the interplay between noise and delay in the vicinity of a fixed point. The DRW representation lends itself to determinations of the joint probability function and, in particular, to the auto-correlation function for both the stationary and the transient states. Thus the effects of delay are manifested through experimentally measurable quantities such as the variance, the correlation time, and the power spectrum. Our findings are illustrated through applications to the analysis of the fluctuations in the center of pressure that occur during quiet standing.

**Keywords:** Delay · Random walk · Stochastic delay differential equation · Fokker–Planck equation · Auto-correlation function · Postural sway

### 11.1 Introduction

Feedback control mechanisms are ubiquitous in physiology [2, 8, 17, 22, 34, 41, 47, 48, 60, 63, 65–67]. There are two important intrinsic features of these control mechanisms: (1) all of them contain time delays and (2) all of them are continually subjected to the effects of random, uncontrolled fluctuations (herein referred to as “noise”). The presence of time delays is a consequence of the simple fact that the different sensors that detect changes in the controlled variable and the effectors that act on this variable are spatially distributed. Since transmission and conduction times are finite, time delays are unavoidable. As a consequence, mathematical models for feedback control take the form of stochastic delay differential equations (SDDE);

an example is the delayed Langevin equation or first-order SDDE with additive noise [20, 25, 33, 36, 37, 42, 43, 53, 59]

$$dx(t) = -kx(t - \tau)dt + dW, \quad (11.1)$$

where  $x(t), x(t - \tau)$  are, respectively, the values of the state variable at times  $t$ , and  $t - \tau$ ,  $\tau$  is the time delay,  $k$  is a constant, and  $W$  describes the Wiener process. In order to obtain a solution of (11.1) it is necessary to define an initial function,  $x(t) = \Phi(t), t \in [-\tau, 0]$ , denoted herein as  $\Phi_0(t)$ .

Understanding the properties of SDDEs is an important first step for interpreting the nature of the fluctuations in physiological variables measured experimentally [11, 37, 53]. However, an increasingly popular way to analyze these fluctuations has been to replace the SDDE by a delayed random walk, i.e., a discrete random walk for which the transition probabilities at the  $n$ -th step depend on the position of the walker  $\tau$  steps before [55–59]. Examples include human postural sway [49, 56], eye movements [46], neolithic transitions [18], econophysics [23, 58], and stochastic resonance-like phenomena [57]. What is the proper way to formulate the delayed random walk so that its properties are equivalent to those predicted by (11.1)?

An extensive mathematical literature has been devoted to addressing issues related to the existence and uniqueness of solutions of (11.1) and their stability [51, 52]. These fundamental mathematical studies have formed the basis for engineering control theoretic studies of the effects of the interplay between noise and delay on the stability of man-made feedback control mechanisms [6, 54]. Lost in these mathematical discussions of SDDEs is the fact that the nearly 100 years of careful experimental observation and physical insight that established the correspondence between (11.1) and an appropriately formulated random walk when  $\tau = 0$  [15, 16, 21, 31, 40, 45, 61] does not have its counterpart for the case when  $\tau \neq 0$ . Briefly the current state of affairs is as follows. The continuous time model described by (11.1), referred to as the delayed Langevin equation, and the delayed random walk must be linked by a Fokker–Planck equation; i.e. a partial differential equation which describes the time evolution of the probability density function. This is because all of these models describe the same phenomenon and hence they must be equivalent in some sense. When  $\tau = 0$  it has been well demonstrated that the Langevin equation and the random walk leads to the same Fokker–Planck equation provided that the random walk occurs in a harmonic, or quadratic, potential (the Ehrenfest model) [31]. Although it has been possible to derive the Fokker–Planck equation from (11.1) when  $\tau \neq 0$  [19, 59], the form of the Fokker–Planck equation obtained from the delayed random walk has not yet been obtained. One of the objectives of this chapter is to show that the Fokker–Planck equation for the random walk can be readily obtained by generalizing the Ehrenfest model on a quadratic potential to non-zero delay. The importance of this demonstration is that it establishes that (11.1) and this delayed random walk give two different, but complimentary views of the same process.

Since (11.1) indicates that the dynamics observed at time  $t$  depend on what happened at time  $t - \tau$ , it is obvious that the joint probability function must play a fundamental role in understanding the interplay between noise and delay. Moreover, the auto-correlation function,  $c(\Delta) \equiv \langle x(t)x(t + \Delta) \rangle$ , is essential for the experimental descriptions of real dynamical systems [4, 14, 30]. This follows from the fact that

three measurements are required to fully describe a noisy signal: (1) its probability density function (e.g., uniform, Gaussian); (2) its intensity; and (3) its correlation time (e.g., white, colored). From a knowledge of  $c(\Delta)$  we can obtain an estimate of the variance ( $\Delta = 0$ ) which provides a measure of signal intensity, the correlation time, and the power spectrum. Armed with these quantities the experimentalist can directly compare experimental observation with prediction. Surprisingly little attention has been devoted to the subject of the joint probability functions in the SDDE literature.

The organization of this chapter is as follows. First, we review the simple random walk that appears in standard introductory textbooks [1,5,40,45,62]. We use this simple random walk to introduce a variety of techniques that are used in the subsequent discussion including the concepts of the generating and characteristic functions, joint probability, and the inter-relationship between the auto-correlation function and the power spectral density (Wiener–Khintchine theorem). The Fokker–Planck equation for the simple random walk is the familiar diffusion equation [45]. Second, we discuss the Ehrenfest model for a discrete random walk in a quadratic, or harmonic, potential in order to introduce the concept of stability into a random walk. Third, we introduce a discrete delay into the Ehrenfest model. The Fokker–Planck equation is obtained and is shown to be identical to that obtained starting from (11.1). In all cases, particular attention is given to obtaining an estimate of  $c(\Delta)$  and to demonstrating how the presence of  $\tau$  influences the correlation time and the power spectrum. In the final section, we review the application of delayed random walk models to the analysis of the fluctuations recorded during human postural sway.

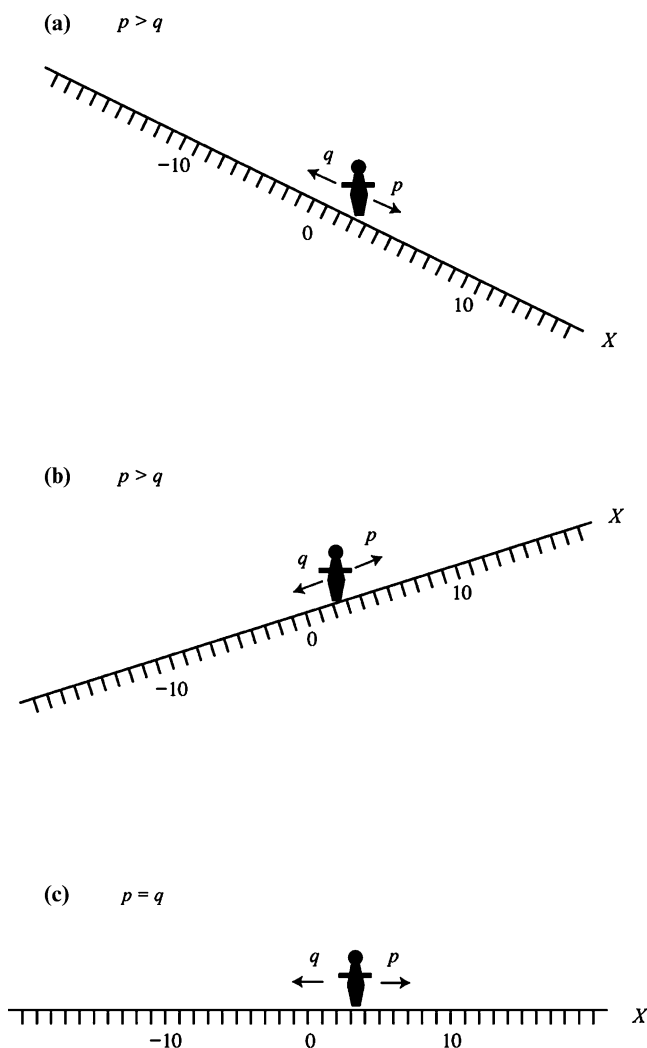
## 11.2 Simple Random Walk

Analyses of random walks in their various forms lie at the core of statistical physics [16,40,45,61,64] and their applications ranging from biology [5] to economics [44]. The simplest case describes a walker who is confined to move along a line by taking identical discrete steps at identical discrete time intervals (Fig. 11.1). Let  $X(n)$  be the position of the walker after the  $n$ -th step. Assume that at zero time all walkers start at the origin, i.e., the initial condition is  $X(0) = 0$ , and that the probability,  $p$ , that the walker takes a step of unit length,  $\ell$ , to the right (i.e.,  $X(n+1) - X(n) = +\ell$ ) is the same as the probability,  $q$ , that it takes a step of unit length to the left (i.e.,  $X(n+1) - X(n) = -\ell$ ), i.e.,  $p = q = 0.5$ .

The total displacement,  $X$ , after  $n$  steps is

$$X = \sum_{i=1}^n \ell_i, \quad (11.2)$$

where  $\ell = \pm 1$ . Since steps to the right and to the left are equally probable, then after a large number of steps, we have



**Fig. 11.1** Conceptual view of simple random walks. The probability to take a step to the right is  $p$ ; a step to the left is  $q$ : (a)  $p > q$ ; (b)  $p < q$ ; and (c)  $p = q = 0.5$

$$\langle X \rangle = \sum_{i=1}^n \langle \ell_i \rangle = 0, \quad (11.3)$$

where the notation  $\langle \dots \rangle$  signifies the “ensemble” average.

Of course each time we construct, or realize, a particular random walk in this manner, the relationship given by (11.3) provides us no information as to how far a given walker is displaced from the origin after  $n$  steps. One way to consider the displacement of each realization of a random walk is to compute  $X^2$ , i.e.,

$$X^2 = (\ell_1 + \ell_2 + \cdots + \ell_n)(\ell_1 + \ell_2 + \cdots + \ell_n) \quad (11.4)$$

$$= \sum_{i=1}^n \ell_i^2 + \sum_{i \neq j}^n \ell_i \ell_j. \quad (11.5)$$

If we now average over many realizations of the random walk we obtain

$$\langle X^2 \rangle = \sum_{i=1}^n \langle \ell_i^2 \rangle + \sum_{i \neq j}^n \langle \ell_i \ell_j \rangle. \quad (11.6)$$

The first term is obviously  $n\ell^2$ . The second term vanishes since the direction of the step that a walker takes at a given instance does not depend on the direction of previous steps. In other words, the direction of steps taken by the walker are uncorrelated and  $\ell_i$  and  $\ell_j$  are independent for  $i \neq j$ . Hence we have

$$\langle X^2 \rangle = n\ell^2. \quad (11.7)$$

The problem with this method of analysis of the random walk is that it is not readily transferable to more complex types of random walks. In particular, in order to use the notion of a random walk to investigate the properties of a stochastic delay differential equations, such as (11.1), it is necessary to introduce more powerful tools such as the characteristic, generating, and auto-correlation functions. Without loss of generality we assume that the walker takes a step of unit length, i.e.,  $|\ell| = 1$ .

### 11.2.1 Probability Distribution Function

The probability that after  $n$  steps the walker attains a position  $r$  ( $X(n) = r$ ) is  $P(X = r, n) = P(r, n)$ , where  $P(r, n)$  is the probability distribution function and satisfies

$$\sum_{r=-\infty}^{\infty} P(r, n) = 1.$$

By analogy with the use of the characteristic function in continuous dynamical systems [7, 14], the discrete characteristic function,

$$R(\theta, n) = \sum_{r=-\infty}^{+\infty} P(r, n) e^{j\theta r}, \quad (11.8)$$

where  $\theta$  is the continuous “frequency” parameter, can be used to calculate  $P(r, n)$ . Since  $R(\theta, n)$  is defined in terms of a Fourier series whose coefficients are the  $P(r, n)$ , the probability distribution after  $n$  steps can be represented in integral form as

$$P(r, n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} R(\theta, n) e^{-j\theta r} d\theta. \quad (11.9)$$

In order to use  $R(\theta, n)$  to calculate  $P(r, n)$  we first write down an equation that describes the dynamics of the changes in  $P(r, n)$  as a function of the number of steps, i.e.,

$$\begin{aligned} P(r, 0) &= \delta_{r,0}, \\ P(r, n) &= pP(r-1, n-1) + qP(r+1, n-1), \end{aligned} \quad (11.10)$$

where  $\delta_{r,0}$  is the Kronecker delta function defined by

$$\begin{aligned} \delta_{r,0} &= 1, \quad (r=0), \\ \delta_{r,0} &= 0, \quad (r \neq 0). \end{aligned}$$

Second, we multiply both sides of (11.10) by  $e^{j\theta r}$ , and sum over  $r$  to obtain

$$\begin{aligned} R(\theta, 0) &= 1 \\ R(\theta, n) &= (pe^{j\theta} + qe^{-j\theta})R(\theta, n-1). \end{aligned}$$

The solution of these equations is

$$R(\theta, n) = (pe^{j\theta} + qe^{-j\theta})^n \quad (11.11)$$

Taking the inverse Fourier transform of (11.11) we eventually obtain

$$\begin{aligned} P(r, n) &= \binom{n}{\frac{r+n}{2}} p^{\frac{r+n}{2}} q^{\frac{n-r}{2}}, \quad (r+n=2m) \\ &= 0, \quad (r+n \neq 2m), \end{aligned} \quad (11.12)$$

where  $m$  is a nonnegative integer. For the special case that  $p = q = 0.5$  this expression for  $P(r, n)$  simplifies to

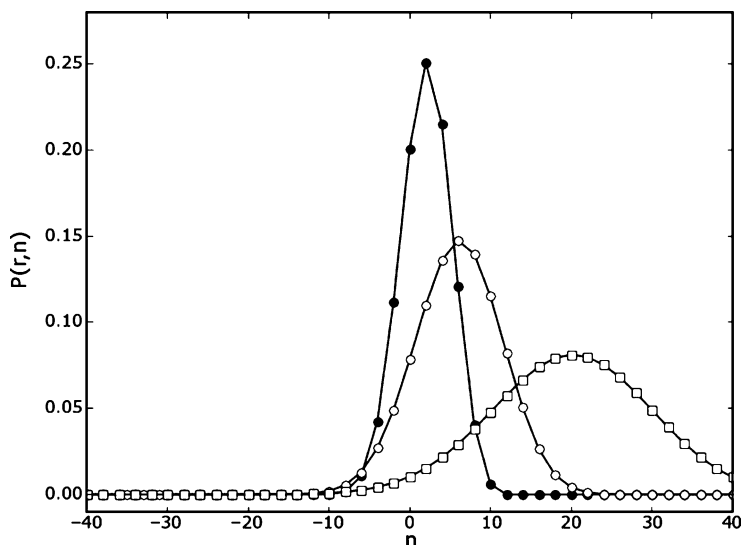
$$\begin{aligned} P(r, n) &= \binom{n}{\frac{r+n}{2}} \left(\frac{1}{2}\right)^n, \quad (r+n=2m) \\ &= 0, \quad (r+n \neq 2m), \end{aligned} \quad (11.13)$$

and we obtain

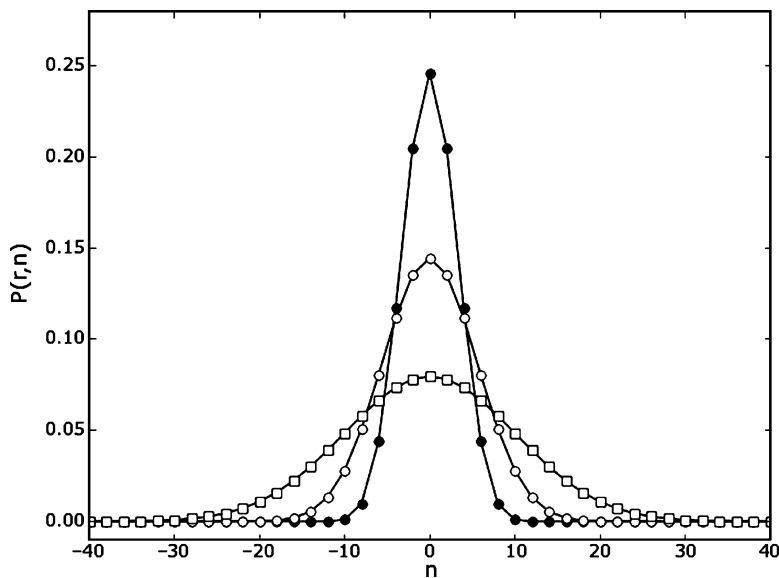
$$\langle X(n) \rangle = 0, \quad (11.14)$$

$$\sigma^2(n) = n. \quad (11.15)$$

When  $p > q$  the walker drifts toward the right (Fig. 11.1a) and when  $p < q$  toward the left (Fig. 11.1b). The evolution of  $P(r, n)$  as a function of time when  $p = 0.6$  is shown in Fig. 11.2 (for the special case of Brownian motion, i.e.,  $p = q = 0.5$ , see Fig. 11.3).



**Fig. 11.2** The probability density function,  $P(r,n)$ , as a function of time for a simple random walk that starts at the origin with  $p = 0.6$ : (closed circle)  $n = 10$ , (open circle)  $n = 30$ , (open square)  $n = 100$ . We have plotted  $P(r,n)$  for even  $r$  only: for these values of  $n$ ,  $P(r,n) = 0$  when  $r$  is odd



**Fig. 11.3** The probability density function,  $P(r,n)$ , as a function of time for Brownian motion, i.e., a simple random walk that starts at the origin with  $p = 0.5$ : (closed circle)  $n = 10$ , (open circle)  $n = 30$ , (open square)  $n = 100$ . We have plotted  $P(r,n)$  for even  $n$  only: for these values of  $n$ ,  $P(r,n) = 0$  when  $n$  is odd

### 11.2.2 Variance

The importance of  $P(r, n)$  is that averaged quantities of interest, or expectations, such as the mean and variance can be readily determined from it. For a discrete variable,  $X$ , the moments can be determined by using the generating function,  $Q(s, n)$ , i.e.,

$$Q(s, n) = \sum_{r=-\infty}^{+\infty} s^r P(r, n). \quad (11.16)$$

The averaged quantities of interest are calculated by differentiating  $Q(s, n)$  with respect to  $s$ .

By repeating arguments analogous to those we used to obtain  $P(r, n)$  from the characteristic function, we obtain

$$Q(s, n) = \left[ ps + \frac{q}{s} \right]^n. \quad (11.17)$$

The mean is obtained by differentiating  $Q(s, n)$  with respect to  $s$

$$\frac{\partial}{\partial s} Q(s, n) \big|_{s=1} \equiv \langle X(n) \rangle = n(p - q). \quad (11.18)$$

The second differentiation leads to the variance

$$\frac{\partial^2}{\partial s^2} Q(s, n) \big|_{s=1} = \langle X(n)^2 \rangle - \langle X(n) \rangle^2 \equiv \sigma^2(n) = 4npq. \quad (11.19)$$

The variance can be calculated from these two equations as  $\sigma^2(n) = 4npq$ .

The variance gives the intensity of the varying component of the random process, i.e., the AC component or the variance. The positive square root of the variance is the standard deviation which is typically referred to as the root-mean-square (rms) value of the AC component of the random process. When  $\langle X(n) \rangle = 0$ , the variance equals the mean square displacement.

### 11.2.3 Fokker–Planck Equation

We note here briefly that the random walks presented here has a correspondence with the continuous time stochastic partial differential equation

$$\frac{dx}{dt} = \mu + \xi(t), \quad (11.20)$$

where  $\xi(t)$  is a gaussian white noise and  $\mu$  is a “drift constant.” Both from the random walk presented above and from this differential equation, we can obtain the Fokker–Planck equation [21, 45].

$$\frac{\partial}{\partial t}P(x,t) = -v\frac{\partial}{\partial x}P(x,t) + D\frac{\partial^2}{\partial x^2}P(x,t), \quad (11.21)$$

where  $v$  and  $D$  are constants. When there is no bias,  $\mu = 0$ ,  $p = q = 1/2$  and we have  $v = 0$ , leading to the diffusion equation. This establishes a link between the Wiener process and simple symmetric random walk.

### 11.2.4 Auto-correlation Function: Special Case

The stationary discrete auto-correlation function,  $C(\Delta)$ , provides a measure of how much average influence random variables separated  $\Delta$  steps apart have on each other. Typically little attention is given to determining the auto-correlation function for a simple random walk. However, it is useful to have a baseline knowledge of what the auto-correlation looks like for a simple random process.

Suppose we measure the direction that the simple random walker moves each step. Designate a step to the right as  $R$ , a step to the left as  $L$ , and a “flip” as an abrupt change in the direction that the walker moves. Then the time series for a simple random walker takes the form

$$R \cdots \overbrace{RL}^{\text{flip}} \cdots \underbrace{LR}_{\text{flip}} \cdots \overbrace{RL}^{\text{flip}} \cdots \quad (11.22)$$

Assume that the step interval,  $\delta n$ , is so small that the probability that two flips occur within the same step is approximately zero. The auto-correlation function,  $C(\Delta)$ , where  $|\Delta| \geq |\delta n|$ , for this process will be

$$C(\Delta) \equiv \langle X(n)X(n+|\Delta|) \rangle = A^2(p_0(\Delta) - p_1(\Delta) + p_2(\Delta) - p_3(\Delta) + \cdots), \quad (11.23)$$

where  $p_\kappa(\Delta)$  is the probability that in a time interval  $\Delta$  that exactly  $\kappa$  flips occur and  $A$  is the length of each step.

In order to calculate the  $p_\kappa$  we proceed as follows. The probability that a flip occurs in  $\delta n$  is  $\lambda \delta n$ , where  $\lambda$  is some suitably defined parameter. Hence the probability that no flip occurs is  $1 - \lambda \delta n$ . If  $n > 0$  then the state involving precisely  $\kappa$  flips in the interval  $(n, n + \delta n)$  arises from either  $\kappa - 1$  events in the interval  $(0, n)$  with one flip in time  $\delta n$ , or from  $\kappa$  events in the interval  $(0, n)$  and no new flips in  $\delta n$ . Thus

$$p_\kappa(n + \delta n) = p_{\kappa-1}(n)\lambda \delta n + p_\kappa(n)(1 - \lambda \delta n) \quad (11.24)$$

and hence we have

$$\lim_{\delta n \rightarrow 0} \frac{p_\kappa(n + \delta n) - p_\kappa(n)}{\delta n} \equiv \frac{dp_\kappa(n)}{dn} = \lambda [p_{\kappa-1}(n) - p_\kappa(n)] \quad (11.25)$$

for  $\kappa > 0$ . When  $\kappa = 0$  we have

$$\frac{dp_0(n)}{dn} = -\lambda p_0(n) \quad (11.26)$$

and at  $n = 0$

$$p_0(0) = 1. \quad (11.27)$$

Equations (11.25)–(11.27) describe an iterative procedure to determine  $p_\kappa$ . In particular we have

$$p_\kappa(\Delta) = \frac{(\lambda|\Delta|)^\kappa e^{-\lambda|\Delta|}}{\kappa!}. \quad (11.28)$$

The required auto-correlation function  $C(\Delta)$  is obtained by combining (11.23) and (11.28) as

$$C(\Delta) = A^2 e^{-2\lambda|\Delta|}. \quad (11.29)$$

The auto-correlation function,  $C(\Delta)$ , and the power spectrum,  $W(f)$ , are intimately connected. In particular, they form a Fourier transform pair

$$W(f) = \Delta \sum_{m=-(n-1)}^{n-1} C(\Delta) e^{-j2\pi f m \Delta}, \quad -\frac{1}{2\Delta} \leq f < \frac{1}{2\Delta} \quad (11.30)$$

and

$$C(\Delta) = \int_{-1/2\Delta}^{1/2\Delta} W(f) e^{j2\pi f \Delta} df, \quad -N\Delta \leq \Delta \leq N\Delta. \quad (11.31)$$

Our interest is to compare  $C(\Delta)$  and  $W(f)$  calculated for a discrete random walk to those that would be observed for a continuous random walk, respectively,  $c(\Delta)$  and  $w(f)$ . Thus we assume in the discussion that follows that the length of the step taken by the random walker can be small enough so that we can replace (11.30) and (11.31) by, respectively,

$$w(f) = \int_{-\infty}^{\infty} c(\Delta) e^{-j2\pi f \Delta} d\Delta \quad (11.32)$$

and

$$c(\Delta) = \int_{-\infty}^{\infty} w(f) e^{j2\pi f \Delta} df. \quad (11.33)$$

Together (11.32) and (11.33) (or (11.30) and (11.31) are referred to as the *Wiener–Khinchine theorem*.

The power spectrum,  $w(f)$ , describes how the energy (or variance) of signal is distributed with respect to frequency. Since the energy must be the same whether we are in the time or frequency domain, it must necessarily be true that

$$\int_{-\infty}^{\infty} |X(t)|^2 dt = \int_{-\infty}^{\infty} |w(f)|^2 df. \quad (11.34)$$

This result is sometimes referred to as Parseval's formula. If  $g(t)$  is a continuous signal,  $w(f)$  of the signal is the square of the magnitude of the continuous Fourier transform of the signal, i.e.,

$$w(f) = \left| \int_{-\infty}^{\infty} g(t) e^{-j2\pi ft} dt \right|^2 = |G(f)|^2 = G(f)G^*(f), \quad (11.35)$$

where  $G(f)$  is the continuous Fourier transform of  $g(t)$  and  $G^*(f)$  is its complex conjugate.

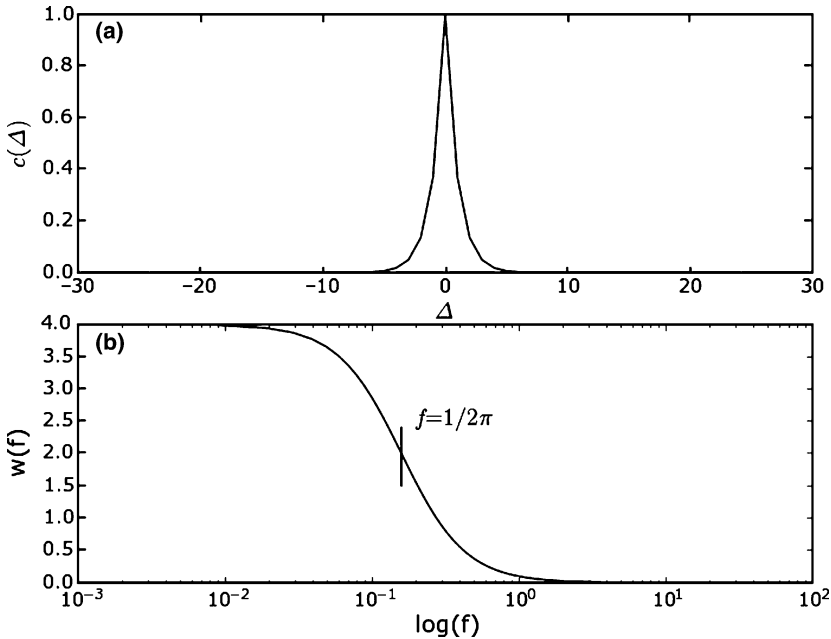
Thus we can determine  $w(f)$  for this random process as

$$w(f) = \frac{2A^2}{\lambda} \left[ \frac{1}{1 + (\pi f/\lambda)^2} \right]. \quad (11.36)$$

Figure 11.4 shows  $c(\Delta)$  and  $w(f)$  for this random process, where  $f$  is  $\Delta^{-1}$ . We can see that the noise spectrum for this random process is essentially “flat” for  $f \ll \lambda$  and thereafter decays rapidly to zero with a power law  $1/f^2$ .

### 11.2.5 Auto-correlation Function: General Case

The important point is that the Fourier transform pairs given by (11.32) and (11.33) are valid whether  $X(t)$  represents a deterministic time series or a realization of a stochastic process [30]. Unfortunately, it is not possible to reliably estimate  $w(f)$  from a finitely long time series (see pp. 211–213 in [30]). The problem is that



**Fig. 11.4** (a) Auto-correlation function,  $c(\Delta)$  and (b) power spectrum,  $w(f)$ , for the random process described by (11.22). Parameters:  $A = 1$ ,  $\lambda = 0.5$

$w(f)$  measured for a finite length stochastic processes does not converge in any statistical sense to a limiting value as the sample length becomes infinitely long. This observation should not be particularly surprising. Fourier analysis is based on the assumption of fixed amplitudes, frequencies, and phases, but time series of stochastic processes are characterized by random changes of amplitudes, frequencies, and phases. Unlike  $w(f)$  a reliable estimate of  $c(\Delta)$  can be obtained from a long time series. These observations emphasize the critical importance of the Wiener–Khinchine theorem for describing the properties of a stochastic dynamical system in the frequency domain.

In order to define  $C(\Delta)$  for a random walk it is necessary to introduce the concept of a joint probability,  $P(r, n_1; m, n_2)$ . The joint probability is a probability density function that describes the occurrence  $X(n_1) = r$  and  $X(n_2) = m$ , i.e. the probability that  $X(n_1)$  is at  $r$  at  $n_1$  when  $X(n_2)$  is at site  $m$  at  $n_2$ . The importance of  $P(r, n_1; m, n_2)$  is that we can use it to determine the probability that each of these events occurs separately. In particular, the probability,  $P(r, n_1)$ , that  $X(n_1)$  is at site  $r$  irrespective of the location of  $X(n_2)$  is

$$P(r, n_1) = \sum_{m=-\infty}^{+\infty} P(r, n_1; m, n_2).$$

Similarly, the probability,  $P(m, n_2)$ , that  $X(n_2)$  is at site  $m$  irrespective of the location of  $X(n_1)$  is

$$P(m, n_2) = \sum_{r=-\infty}^{+\infty} P(r, n_1; m, n_2).$$

Using these definitions and the fact that  $P(r, n_1; m, n_2)$  is a joint probability distribution function, we can determine  $C(\Delta, n)$  to be

$$\begin{aligned} C(\Delta, n) &\equiv \langle X(n)X(n-\Delta) \rangle \\ &= \sum_{r=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} rmP(r, n; m, n-\Delta). \end{aligned} \quad (11.37)$$

The stationary auto-correlation function can be defined by taking the long time limit as follows

$$\begin{aligned} C(\Delta) &\equiv \langle X(n)X(n-\Delta) \rangle_s \equiv \lim_{n \rightarrow \infty} C(\Delta, n) \\ &= \lim_{n \rightarrow \infty} \sum_{r=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} rmP(r, n; m, n-\Delta). \end{aligned} \quad (11.38)$$

### 11.3 Random Walks on a Quadratic Potential

The major limitation for applying simple random walk models to questions related to feedback is that they lack the notion of *stability*, i.e., the resistance of dynamical systems to the effects of perturbations. In order to understand how stability can be

incorporated into a random walk, it is useful to review the concept of a potential function,  $\phi(x)$ , in continuous time dynamical systems. For  $\tau = 0$  the deterministic version of (11.1) can be written as [26]

$$\dot{x}(t) = -kx(t) = -\frac{d\phi}{dx} \quad (11.39)$$

and hence

$$\phi(x) = \int_0^x g(s)ds = \frac{kx^2}{2}$$

describes a quadratic, or harmonic, function. The bottom of this well corresponds to the fixed-point attractor  $x = 0$ ; the well is its basin of attraction. If  $x(t)$  is a solution of (11.39) then

$$\begin{aligned} \frac{d}{dt}\phi(x(t)) &= \frac{d}{dx}\phi(x(t)) \cdot \frac{d}{dt}x(t) \\ &= -[g(x(t))]^2 \leq 0. \end{aligned}$$

In other words  $\phi$  is always decreasing along the solution curves and in this sense is analogous to the “potential functions” in physical systems.

The urn model developed by Paul and Tatyana Ehrenfest showed how a quadratic potential could be incorporated into a discrete random walk [15, 31, 32]. The demonstration that the Fokker–Planck equation obtained for the Ehrenfest random walk is the same as that for Langevin’s equation, i.e., (11.1), is due to Kac [31]. Here we derive the auto-correlation function,  $C(\Delta)$ , for the Ehrenfest random walk and the Langevin equation. The derivation of the Fokker–Planck equation for  $\tau = 0$  and  $\tau \neq 0$  is identical. Therefore we present the Fokker–Planck equation for the Ehrenfest random walk as a special case of that for a delayed random walk in Sect. 11.4.1.

By analogy with the above observations, we can incorporate the influence of a quadratic-shaped potential on a random walker by assuming that the transition probability toward the origin increases linearly with distance from the origin (of course up to a point). In particular, the transition probability for the walker to move toward the origin increases linearly at a rate of  $\beta$  as the distance increases from the origin up to the position  $\pm a$  beyond which it is constant (since the transition probability is between 0 and 1). Equation (11.10) becomes

$$\begin{aligned} P(r, 0) &= \delta_{r,0}, \\ P(r, n) &= g(r-1)P(r-1, n-1) \\ &\quad + f(r+1)P(r+1, n-1), \end{aligned} \quad (11.40)$$

where  $a$  and  $d$  are positive parameters,  $\beta = 2d/a$ , and

$$f(x) = \begin{cases} \frac{1+2d}{2} & x > a \\ \frac{1+\beta n}{2} & -a \leq x \leq a, \\ \frac{1-2d}{2} & x < -a \end{cases}$$

$$g(x) = \begin{cases} \frac{1-2d}{2} & x > a \\ \frac{1-\beta n}{2} & -a \leq x \leq a \\ \frac{1+2d}{2} & x < -a, \end{cases}$$

where  $f(x), g(x)$  are, respectively, the transition probabilities to take a step in the negative and positive directions at position  $x$  such that

$$f(x) + g(x) = 1. \quad (11.41)$$

The random walk is symmetric with respect to the origin provided that

$$f(-x) = g(x) \quad (\forall x). \quad (11.42)$$

We classify random walks by their tendency to move toward the origin. The random walk is said to be *attractive* when

$$f(x) > g(x) \quad (x > 0). \quad (11.43)$$

and *repulsive* when

$$f(x) < g(x) \quad (x > 0). \quad (11.44)$$

We note that when

$$f(x) = g(x) = \frac{1}{2} \quad (\forall x), \quad (11.45)$$

the general random walk given by (11.41) reduces to the simple random walk discussed Sect. 11.3. In this section, we consider only the attractive case.

### 11.3.1 Auto-correlation Function: Ehrenfest Random Walk

Assume that with sufficiently large  $a$ , we can ignore the probability that the walker is outside of the range  $(-a, a)$ . In this case, the probability distribution function  $P(r, n)$  approximately satisfies the equation

$$\begin{aligned} P(r, n) = & \frac{1}{2}(1 - \beta(r-1))P(r-1, n-1) \\ & + \frac{1}{2}(1 + \beta(r+1))P(r+1, n-1). \end{aligned} \quad (11.46)$$

By symmetry, we have

$$\begin{aligned} P(r, n) &= P(-r, n), \\ \langle X(t) \rangle &= 0. \end{aligned}$$

The variance, obtained by multiplying (11.46) by  $r^2$  and summing over all  $r$ , is

$$\sigma^2(n) = \langle X^2(n) \rangle = \frac{1}{2\beta}(1 - (1 - 2\beta)^n). \quad (11.47)$$

Thus the variance in the stationary state is

$$\sigma_s^2 = \langle X^2 \rangle_s = \frac{1}{2\beta}. \quad (11.48)$$

The auto-correlation function for this random walk can be obtained by rewriting (11.46) in terms of joint probabilities to obtain

$$\begin{aligned} P(r, n; m, n - \Delta) &= \frac{1}{2}(1 - \beta(r - 1))P(r - 1, n - 1; m, n - \Delta) \\ &+ \frac{1}{2}(1 + \beta(r + 1))P(r + 1, n - 1; m, n - \Delta). \end{aligned} \quad (11.49)$$

By defining

$$P_s(r; m, \Delta) \equiv \lim_{n \rightarrow \infty} P(r, n; m, n - \Delta)$$

the joint probability obtained in the stationary state, i.e., the long time limit, becomes

$$\begin{aligned} P_s(r; m, \Delta) &= \frac{1}{2}(1 - \beta(r - 1))P_s(r - 1; m, \Delta - 1) \\ &+ \frac{1}{2}(1 + \beta(r + 1))P_s(r + 1; m, \Delta - 1). \end{aligned} \quad (11.50)$$

Multiplying by  $rm$  and summing over all  $r$  and  $m$  yields

$$C(\Delta) = (1 - \beta)C(\Delta - 1), \quad (11.51)$$

which we can rewrite as

$$C(\Delta) = (1 - \beta)^\Delta C(0) = \frac{1}{2\beta}(1 - \beta)^\Delta, \quad (11.52)$$

where  $C(0)$  is equal to the mean-square displacement,  $\langle X^2 \rangle_s$ . When  $\beta \ll 1$ , we can approximate,

$$C(\Delta) \approx \frac{1}{2\beta} e^{-\beta|\Delta|}. \quad (11.53)$$

Then, from the Wiener–Khinchine theorem we have

$$w(f) \approx \frac{2}{\beta^2} \left[ \frac{1}{1 + (2\pi f/\beta)^2} \right]. \quad (11.54)$$

These expressions for  $K(\Delta)$  and  $w(f)$  are in the same form as those obtained for a random process, i.e., (11.28) and (11.36). Hence  $K(\Delta)$  and  $w(f)$  are qualitatively the same as those shown in Fig. 11.4.

### 11.3.2 Auto-correlation Function: Langevin Equation

When  $\tau = 0$ , (11.1) describes the effects of random perturbations on a dynamical system confined to move within a quadratic potential. It is well known that the variance,  $c(0)$ , is

$$c(0) = \frac{1}{2k}, \quad (11.55)$$

which is identical to (11.53) if we identify  $k$  with  $\beta$  and take  $\Delta = 0$  [21, 35].

In order to determine  $c(\Delta)$  and  $w(f)$  we note that when  $\tau = 0$ , (11.1) describes the effects of a low-pass filter on  $\delta$ -correlated (white) noise. Thus we can write the Fourier transform of (11.1) when  $\tau = 0$  as

$$X(f) = H(f)I(f), \quad (11.56)$$

where the frequency response,  $H(f)$ , is given by

$$H(f) = \frac{1}{j2\pi f + k} \quad (11.57)$$

and

$$I(f) = \sigma^2, \quad (11.58)$$

where  $\sigma$  is a constant. Hence we obtain

$$w(f) = \frac{\sigma^2}{(2\pi f)^2 + k^2} \quad (11.59)$$

and, applying the Weiner–Khinchine theorem,

$$c(\Delta) = \frac{\sigma^2}{2k} e^{-k\Delta}. \quad (11.60)$$

Equation (11.54) is the same as (11.59) and (11.53) is the same as (11.60).

## 11.4 Delayed Random Walks

For a delayed random walk, the transition probability depends on its past state. If we generalize the random walk in a quadratic potential developed in Sect. 11.3 to non-zero delay we obtain the following definition for the transition probability

$$\begin{aligned} P(r, n+1) = & \sum_m g(m) P(r-1, n; m, n-\tau) \\ & + \sum_m f(m) P(r+1, n; m, n-\tau), \end{aligned} \quad (11.61)$$

where the position of the walker at time  $n$  is  $X(n)$ ,  $P(r, n)$  is the joint probability for the walker to be at  $X(n) = r$  and  $P(r, n; m, n - \tau)$  is the joint probability such that  $X(n) = r$  and  $X(n - \tau) = m$  takes place.  $f(x)$  and  $g(x)$  are transition probabilities for the walker to take the step to the negative  $(-1)$  and positive  $(+1)$  directions, respectively, and are the same as those used for the random walk in a quadratic potential described in Sect. 11.3.2.

### 11.4.1 Delayed Fokker–Planck Equation

Here we derive the Fokker–Planck equation for the delayed random walk in a quadratic potential. This method is a direct use of the procedure used by Kac to obtain the Fokker–Planck equation for the case that  $\tau = 0$ . For  $f$  and  $g$  as defined in Sect. 3 (11.61) becomes

$$\begin{aligned} P(r, n+1) &= \sum_m \frac{1}{2} (1 - \beta m) P(r-1, n; m, n - \tau_d) \\ &\quad + \sum_m \frac{1}{2} (1 + \beta m) P(r+1, n; m, n - \tau_d), \end{aligned} \quad (11.62)$$

To make a connection, we treat that the random walk takes a step with a size of  $\Delta x$  at a time interval of  $\Delta t$ , both of which are very small compared to the scale of space and time we are interested in. We take  $x = r\Delta x$  and  $y = m\Delta x$ ,  $t = n\Delta t$  and  $\tau = \tau_d\Delta t$ . With this stipulation, we can rewrite (11.62) as follows

$$\begin{aligned} &\frac{P(x, t + \Delta t) - P(x, t)}{\Delta t} \\ &= \frac{1}{2} \left( \frac{P(x - \Delta x, t) + P(x + \Delta x, t) - 2P(x, t)}{(\Delta x)^2} \right) \left( \frac{(\Delta x)^2}{\Delta t} \right) \\ &\quad + \frac{1}{2} \sum_{\frac{y}{\Delta x}} \frac{y}{\Delta x} (P(x, t; y, t - \tau) - P(x - \Delta x, t; y, t - \tau)) \frac{\beta}{\Delta t} \\ &\quad + \frac{1}{2} \sum_{\frac{y}{\Delta x}} \frac{y}{\Delta x} (P(x + \Delta x, t; y, t - \tau) - P(x, t; y, t - \tau)) \frac{\beta}{\Delta t}. \end{aligned} \quad (11.63)$$

In the limits

$$\begin{aligned} \Delta x &\rightarrow 0, \quad \Delta t \rightarrow 0, \quad \beta \rightarrow 0 \\ \frac{(\Delta x)^2}{2\Delta t} &\rightarrow D, \quad \frac{\beta}{\Delta t} \rightarrow \gamma, \quad n\Delta t \rightarrow t, \quad \tau_d\Delta t \rightarrow \tau \\ n\Delta x &\rightarrow x, \quad m\Delta x \rightarrow y, \end{aligned} \quad (11.64)$$

the difference equation (11.63) goes over to the following integro-partial differential equation

$$\begin{aligned} & \frac{\partial}{\partial t} \int_{-\infty}^{\infty} P(x, t; y, t - \tau) dy \\ &= \int_{-\infty}^{\infty} \gamma \frac{\partial}{\partial x} (y P(x, t; y, t - \tau)) dy + D \int_{-\infty}^{\infty} \frac{\partial^2}{\partial x^2} P(x, t; y, t - \tau) dy. \end{aligned} \quad (11.65)$$

This is the same Fokker–Planck equation that is obtained from the delayed Langevin equation, (11.1) [19]. When  $\tau = 0$  we have

$$P(x, t; y, t - \tau) \rightarrow P(x, t) \delta(x - y), \quad (11.66)$$

and the above Fokker–Planck equation reduces to the following familiar form.

$$\frac{\partial}{\partial t} P(x, t) = \gamma \frac{\partial}{\partial x} (x P(x, t)) + D \frac{\partial^2}{\partial x^2} P(x, t). \quad (11.67)$$

Thus, the correspondence between Ehrenfest's model and Langevin equation carries over to that between the delayed random walk model and the delayed Langevin equation.

We can rewrite (11.65) by making the definition

$$P_{\tau}(x, t) \equiv \int_{-\infty}^{\infty} P(x, t; y, t - \tau) dy$$

to obtain

$$\frac{\partial}{\partial t} P_{\tau}(x, t) = \int_{-\infty}^{\infty} \gamma \frac{\partial}{\partial x} [y P(x, t; y, t - \tau)] dy + D \frac{\partial^2}{\partial x^2} P_{\tau}(x, t). \quad (11.68)$$

In this form we can more clearly see the effect of the delay on the drift of the random walker.

### 11.4.2 Auto-correlation Function: Delayed Random Walk

Three properties of delayed random walks are particularly important for the discussion that follows. First, by the symmetry with respect to the origin, we have that the average position of the walker is 0. In particular, for an attractive delayed random walk, the stationary state (i.e., when  $n \rightarrow \infty$ )

$$P(r, n + 1; r + 1, n) = P(r + 1, n + 1; r, n). \quad (11.69)$$

We can show the above as follows. By the definition of the stationarity, we have

$$P(r, n + 1; r + 1, n) + P(r, n + 1; r - 1, n) \quad (11.70)$$

$$= P(r + 1, n + 1; r, n) + P(r - 1, n + 1; r, n). \quad (11.71)$$

For  $r = 0$ , we note that due to the symmetry, we have

$$P(0, n+1; 1, n) = P(1, n+1; 0, n). \quad (11.72)$$

Using these two equations inductively leads us to the desired relation (11.69).

Second, the generating function

$$\begin{aligned} \langle \cos(\alpha X(n)) \rangle = & \quad (11.73) \\ \cos(\alpha) \langle \cos(\alpha X(n)) \rangle + \sin(\alpha) \langle \sin(\alpha X(n)) \{f(X(n-\tau)) - g(X(n-\tau))\} \rangle \end{aligned}$$

can be obtained by multiplying (11.61) for the stationary state by  $\cos(\alpha r)$  and then summing over  $r$  and  $m$ .

Finally, we have the following invariant relationship with respect to the delay

$$\frac{1}{2} = \langle X(n) \{f(X(n-\tau)) - g(X(n-\tau))\} \rangle. \quad (11.74)$$

When we choose  $f, g$  as before, this invariant relation in (11.74) becomes the following with this model

$$\langle X(n+\tau)X(n) \rangle = C(\tau) = \frac{1}{2\beta}. \quad (11.75)$$

This invariance with respect to  $\tau$  of the correlation function with  $\tau$  steps apart is a simple characteristic of this quadratic potential delayed random walk model. This property is a key to obtaining the analytical expression for the correlation function. Below we discuss  $C(\Delta)$  for the stationary state. Observations pertaining to  $C(\Delta)$  in the transient state are presented in Sect. 11.5.1.

For the stationary state and  $0 \leq \Delta \leq \tau$ , the following is obtained from the definition (11.61):

$$\begin{aligned} P_s(r, n+\Delta; m, n) = & \sum_{\ell} g(\ell) P_s(r-1, n+\Delta; m, n+1; \ell, n+\Delta-\tau) \\ & + \sum_{\ell} f(\ell) P_s(r+1, n+\Delta; m, n+1; \ell, n+\Delta-\tau). \end{aligned} \quad (11.76)$$

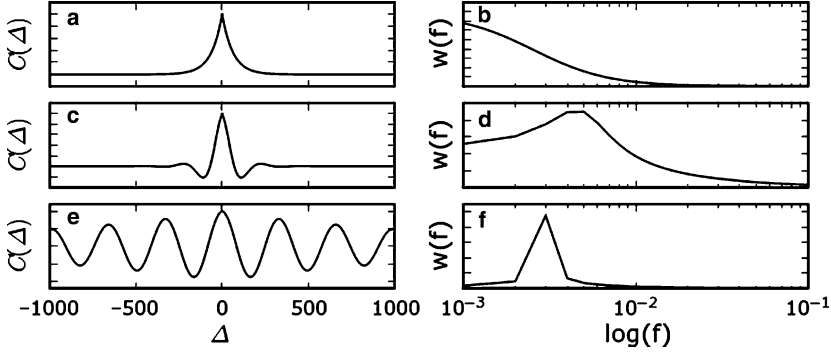
We can derive the following equation for the correlation function by multiplication of this equation by  $rm$  and summing over.

$$C(\Delta) = C(\Delta-1) - \beta C(\tau+1-\Delta), \quad (0 \leq \Delta \leq \tau). \quad (11.77)$$

A similar argument can be given for  $\tau < \Delta$ ,

$$C(\Delta) = C(\Delta-1) - \beta C(\Delta-1-\tau), \quad (\tau < \Delta). \quad (11.78)$$

Equations (11.77) and (11.78) can be solved explicitly using (11.75). In particular, for  $0 \leq \Delta \leq \tau$  we obtain



**Fig. 11.5** Stationary auto-correlation function,  $C(\Delta)$  (left-hand column) and power spectra,  $W(f)$  (right-hand column) for an attractive delayed random walk on a quadratic potential for different time delays,  $\tau$ . Stationary correlation function  $C(u)$  calculated using (11.80) and  $W(f)$  calculated from this using the Wiener–Klitchine theorem. The parameters were  $a = 50$ ,  $d = 0.4$ , and  $\tau$  was 0 for (a) and (b), 40 for (c) and (d) and 80 for (e) and (f)

$$C(\Delta) = C(0) \frac{(z_+^\Delta - z_-^{\Delta-1}) - (z_-^\Delta - z_+^{\Delta-1})}{z_+ - z_-} - \frac{1}{2} \frac{(z_+^\Delta - z_-^\Delta)}{z_+ - z_-},$$

$$C(0) = \frac{1}{2\beta} \frac{(z_+ - z_-) + \beta(z_+^\tau - z_-^\tau)}{(z_+^\tau - z_+^{\tau-1}) - (z_-^\tau - z_-^{\tau-1})}, \quad (11.79)$$

where

$$z_{\pm} = \left(1 - \frac{\beta^2}{2}\right) \pm \frac{\beta}{2} \sqrt{\beta^2 - 4}.$$

For  $\tau < \Delta$ , it is possible to write  $C(\Delta)$  in a multiple summation form, though the expression becomes rather complex. For example, with  $\tau < \Delta \leq 2\tau$ ,

$$C(\Delta) = \frac{1}{2\beta} - \beta \sum_{i=1}^{\Delta-\tau} C(i), \quad (11.80)$$

where the  $C(i)$  are given by (11.79).

Figure 11.5 compares  $C(\Delta)$  and  $W(f)$  for different values of the delay. As we increase  $\tau$ , oscillatory behavior of the correlation function appears. The decay of the peak envelope is found numerically to be exponential. The decay rate of the envelope for the small  $u$  is approximately  $1/(2C(0))$ .

### 11.4.3 Auto-correlation Function: Delayed Langevin Equation

The statistical properties of (11.1) have been extensively studied previously [20, 25, 33, 42, 43]. Two approaches can be used to calculate  $c(\Delta)$ .

First, we can take the Fourier transform of (11.1) to determine  $w(f)$  and then use the Wiener–Khinchine theorem to obtain  $c(\Delta)$ . From (11.56) we obtain the frequency response,  $H(f)$ ,

$$H(f) = \frac{1}{j2\pi f + \mu e^{-j2\pi f\tau}}.$$

As before we assume a “white noise” input and hence the power spectrum,  $w(f)$ , is

$$\begin{aligned} w(f) &= |H(f)\sigma|^2 \\ &= \frac{\sigma^2}{(2\pi f)^2 - 4\pi f\mu \sin 2\pi f\tau + \mu^2}. \end{aligned}$$

Using the Wiener–Khinchine theorem we obtain

$$c(\Delta) = \frac{\sigma^2}{2\pi} \int_0^\infty \frac{\cos \omega \Delta}{\omega^2 - 2\omega\mu \sin(\omega\tau) + \mu^2} d\omega, \quad (11.81)$$

where we have defined  $\omega = 2\pi f$  to simplify the notation. The variance,  $\sigma_x^2$ , is equal to the value of  $c(\Delta)$  when  $\Delta = 0$ , i.e.,

$$\sigma_x^2 = \frac{\sigma^2}{2\pi} \int_0^\infty \frac{d\omega}{\mu^2 + \omega^2 - 2\mu\omega \sin(\omega\tau)}. \quad (11.82)$$

When (11.82) is integrated numerically the result obtained agrees with the results obtained by K  chler and Mensch (discussed below) for  $0 \leq \Delta \leq \tau$  [25].

Second, K  chler and Mensch showed that the stationary correlation function for  $\Delta < \tau$  could be obtained directly from (11.1) and was equal to

$$c(\Delta) = c(0) \cos(\mu\Delta) - \frac{1}{2\Delta} \sin(\mu\Delta), \quad (11.83)$$

where

$$c(0) = \frac{1 + \sin(\mu\tau)}{2\mu \cos(\mu\tau)}. \quad (11.84)$$

It should be noted that for small delay the variance increases linearly. This observation can be confirmed [25].

We now show that the expressions for  $C(\Delta)$  obtained from the delayed random walk (11.80) is equivalent to those given by (11.83) for  $0 \leq \Delta \leq \tau$ . In particular, for small  $\beta$ , we have

$$\begin{aligned} \frac{(z_+^\Delta - z_+^{\Delta-1}) - (z_-^\Delta - z_-^{\Delta-1})}{z_+ - z_-} &\sim \cos(\beta\Delta), \\ \frac{\beta(z_+^\Delta - z_-^\Delta)}{z_+ - z_-} &\sim \sin(\beta\Delta). \end{aligned}$$

Thus

$$C(\Delta) \sim C(0) \cos(\beta\Delta) - \frac{1}{2\Delta} \sin(\beta\Delta) \quad (11.85)$$

with

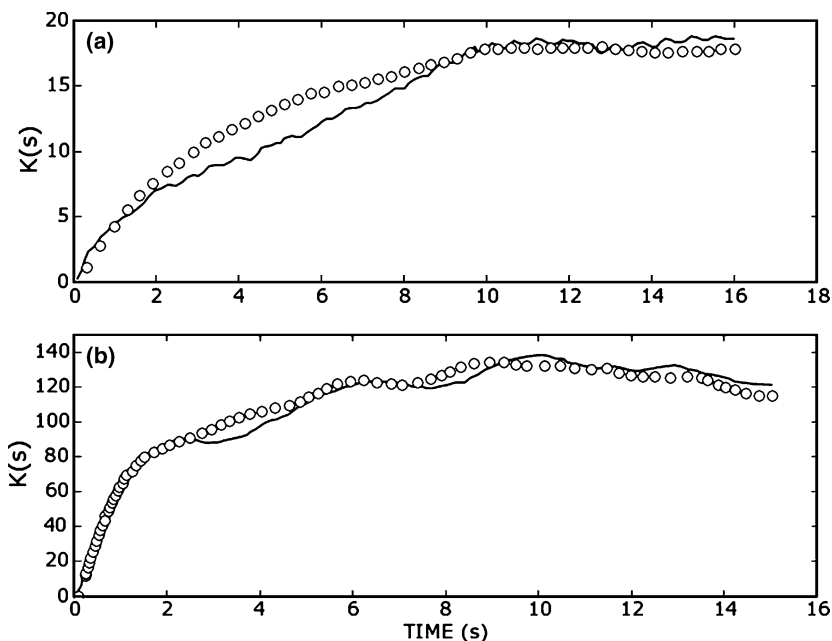
$$C(0) \sim \frac{1 + \sin(\beta\tau)}{2\beta \cos(\beta\tau)}. \quad (11.86)$$

## 11.5 Postural Sway

Postural sway refers to the fluctuations in the center of pressure (COP) that occur as a subject stands quietly with eyes closed on a force platform [13, 53]. Mathematical models for balance control identify three essential components [17, 49, 50, 56]: (1) feedback, (2) time delays, and (3) the effects of random, uncontrolled perturbations (“noise”). Thus it is not surprising that the first application of a delayed random walk was to investigate the fluctuations in COP [56]. In this study it was assumed that the probability  $p_+(n)$  for the walker to take a step at time  $n$  to the right (positive direction) was given by

$$p_+(n) = \begin{cases} p & X(n-\tau) > 0 \\ 0.5 & X(n-\tau) = 0 \\ 1-p & X(n-\tau) < 0, \end{cases} \quad (11.87)$$

where  $0 < p < 1$ . The origin is attractive when  $p < 0.5$ . By symmetry with respect to the origin we have  $\langle X(n) \rangle = 0$ . As shown in Fig. 11.6 this simple model was remarkably capable of reproducing some of the features observed for the fluctuations in the COP observed for certain human subjects.



**Fig. 11.6** Comparison of the two-point correlation function,  $K(s) = \langle (X(n) - X(n-s))^2 \rangle$ , for the fluctuations in the center-of-pressure observed for two healthy subjects (*solid line*) with that predicted using a delayed random walk model (*open circle*). In (a) the parameters for the delayed random walk were  $p = 0.35$  and  $\tau = 1$  with an estimated unit step length of 1.2 mm and a unit time of 320 ms. In (b) the parameters were  $p = 0.40$  and  $\tau = 10$  with an estimated step length and unit time step of, respectively, 1.4 mm and 40 ms. For more details see [56]

There are a number of interesting properties of this random walk (Fig. 11.7). First, for all choices of  $\tau \geq 0$ ,  $\sqrt{\langle X^2(n) \rangle}$  approaches a limiting value,  $\Psi$ . Second, the qualitative nature of the approach of  $\sqrt{\langle X^2(n) \rangle}$  to  $\Psi$  depends on the value of  $\tau$ . In particular, for short  $\tau$  there is a nonoscillatory approach to  $\Psi$ , whereas for longer  $\tau$  damped oscillations occur (Fig. 11.7) whose period is approximately twice the delay. Numerical simulations of this random walk led to the approximation

$$\Psi(\tau) \sim (0.59 - 1.18p)\tau + \frac{1}{\sqrt{2(1-2p)}}. \quad (11.88)$$

This approximation was used to fit the delayed random walk model to the experimentally measured fluctuations in postural sway shown in Fig. 11.6.

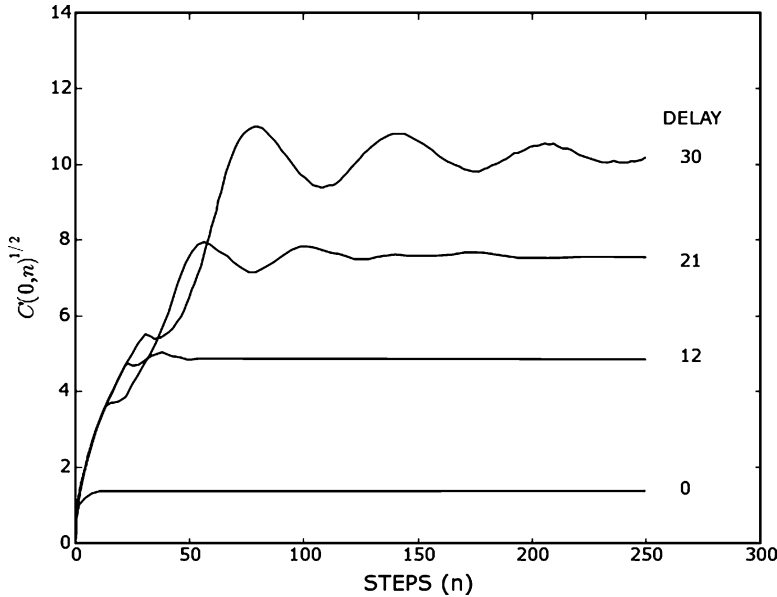
In the context of a generalized delayed random walk (11.61) introduced in Sect. 11.4, (11.87) corresponds to choosing  $f(x)$  and  $g(x)$  to be

$$f(x) = \frac{1}{2}[1 + \eta\theta(x)], \quad (11.89)$$

$$g(x) = \frac{1}{2}[1 - \eta\theta(x)], \quad (11.90)$$

where

$$\eta = 1 - 2p$$



**Fig. 11.7** Examples of dynamics of the root mean square position  $C(0, t)^{1/2}$  for various choices of  $\tau$  when  $p = 0.25$

and  $\theta$  is a step-function defined by

$$\theta(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0. \end{cases} \quad (11.91)$$

In other words the delayed random walk occurs on a V-shaped potential (which, of course is simply a linear approximation to a quadratic potential). Below we briefly describe the properties of this delayed random walk (for more details see [59]).

By using symmetry arguments it can be shown that the stationary probability distributions  $P_s(X)$  when  $\tau = 0$  can be obtained by solving the system of equations with the long time limit.

$$\begin{aligned} P(0, n+1) &= 2(1-p)P(1, n), \\ P(1, n+1) &= \frac{1}{2}P(0, n) + (1-p)P(2, n), \\ P(r, n+1) &= pP(r-1, n) + (1-p)P(r+1, n) \quad (2 \leq r), \end{aligned} \quad (11.92)$$

where  $P(r, n)$  is the probability to be at position  $r$  at time  $n$ , using the trial function  $P_s(r) = Z^r$ , where

$$P_s(r) = \lim_{n \rightarrow \infty} P(r, n).$$

In this way we obtain

$$\begin{aligned} P_s(0) &= 2C_0p, \\ P_s(r) &= C_0 \left( \frac{p}{1-p} \right)^r \quad (1 \leq r), \end{aligned}$$

where

$$C_0 = \frac{(1-2p)}{4p(1-p)}.$$

Since we know the p.d.f. we can easily calculate the variance when  $\tau = 0$ ,  $\sigma^2(0)$ , as

$$\sigma^2(0) = \frac{1}{2(1-2p)^2}. \quad (11.93)$$

The stationary probability distributions when  $\tau > 0$  can be obtained by solving the set of equations with the long time limit.

for  $(0 \leq r < \tau + 2)$

$$\begin{aligned} P(r, n+1) &= pP(r-1, n; r > 0, n-\tau) + \frac{1}{2}P(r-1, n; r=0, n-\tau) \\ &\quad + (1-p)P(r-1, n; r < 0, n-\tau) + pP(r+1, n; X < 0, n-\tau) \\ &\quad + \frac{1}{2}P(r+1, n; r=0, n-\tau) + (1-p)P(r+1, n; r > 0, n-\tau), \end{aligned}$$

for  $(\tau + 2 \leq r)$

$$P(r, n+1) = pP(r-1, n) + (1-p)P(r+1, n).$$

These equations are very tedious to solve and not very illuminating. Indeed we have only been able to obtain the following results for  $\tau = 1$

$$\langle X^2 \rangle = \frac{1}{2(1-2p)^2} \left( \frac{7-24p+32p^2-16p^3}{3-4p} \right)$$

and for  $\tau = 2$

$$\langle X^2 \rangle = \frac{1}{2(1-2p)^2} \left( \frac{25-94p+96p^2+64p^3-160p^4+64p^5}{5+2p-24p^2+16p^3} \right).$$

### 11.5.1 Transient Auto-correlation Function

In Sect. 11.5, we assumed that the fluctuations in COP were realizations of a stationary stochastic dynamical system. However, this assumption is by no means clear. An advantage of a delayed random walk model is that it is possible to gain some insight into the nature of the auto-correlation function for the transient state,  $C_t(\Delta)$ . In particular, for the transient state we can calculate in the similar manner to (11.77) and (11.78), the set of coupled dynamical equations

$$C_t(0, n+1) = C_t(0, n) + 1 - 2\beta C_t(\tau, n - \tau), \quad (11.94)$$

$$C_t(\Delta, n+1) = C_t(\Delta-1, n+1) - \beta C_t(\tau - (\Delta-1), n + \Delta - \tau), \quad (1 \leq \Delta \leq \tau),$$

$$C_t(\Delta, n+1) = C_t(\Delta-1, n+1) - \beta C_t((\Delta-1) - \tau, n+1), \quad (\Delta > \tau).$$

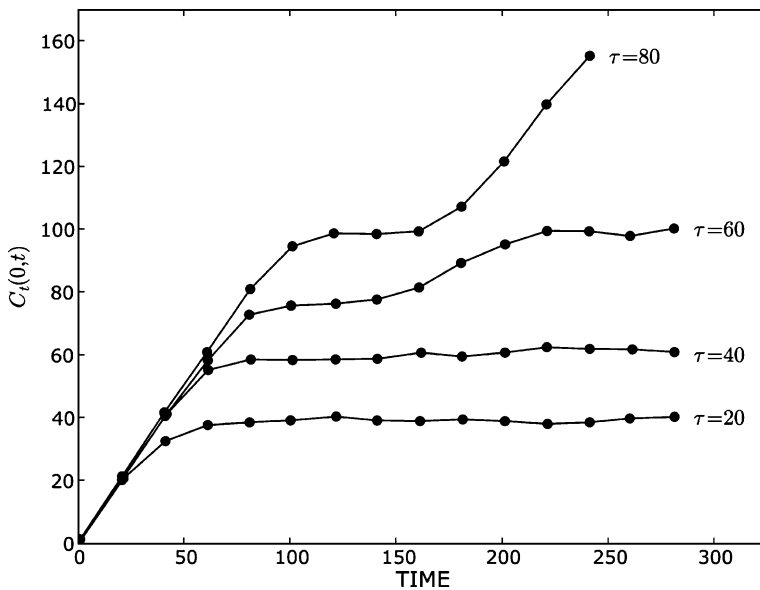
For the initial condition, we need to specify the correlation function for the interval of initial  $\tau$  steps. When the random walker begins at the origin we have a simple symmetric random walk for  $n \in (1, \tau)$ . This translates to the initial condition for the correlation function as

$$C_t(0, n) = n \quad (0 \leq \Delta \leq \tau) \quad C_t(u, 0) = 0 \quad (\forall u). \quad (11.95)$$

The solution can be iteratively generated using (11.95) and this initial condition. We have plotted some examples for the dynamics of the mean square displacement  $C_t(0, n)$  in Fig. 11.8. Again, the oscillatory behavior arises with increasing  $\tau$ . Hence, the model discussed here shows the oscillatory behavior with increasing delay which appears in both its stationary and transient states.

We also note that from (11.95), we can infer the corresponding set of equations for the transient auto-correlation function of (11.1) with a continuous time,  $c_t(\Delta)$ . They are given as follows:

$$\begin{aligned} \frac{\partial}{\partial t} c_t(0, t) &= -2kc_t(\tau, t - \tau) + 1 \\ \frac{\partial}{\partial \Delta} c_t(\Delta, t) &= -kc_t(\tau - \Delta, t + \Delta - \tau) \quad (0 < \Delta \leq \tau) \\ \frac{\partial}{\partial \Delta} c_t(\Delta, t) &= -kc_t(\Delta - \tau, t) \quad (\tau < \Delta) \end{aligned} \quad (11.96)$$



**Fig. 11.8** Examples of dynamics of the transient variance  $C_t(0, t)$  for different delays  $\tau$ . Data averaged from 10,000 simulations (closed circle) is compared to that determined analytically from (11.95) (solid line). The parameters were  $a = 50$  and  $d = 0.45$

Studies on these coupled partial differential equations with delay are yet to be done.

### 11.5.2 Balance Control with Positive Feedback

Up to this point we have assumed that the feedback for balance control is continuous and negative. However, careful experimental observations for postural sway [38, 39] and stick balancing at the fingertip [11, 12, 28] indicate that the feedback is on average positive and that it is administered in a pulsatile, or ballistic, manner. Recently the following switch-type discontinuous model for postural sway that incorporates positive feedback has been introduced in an attempt to resolve this paradox [49, 50]:

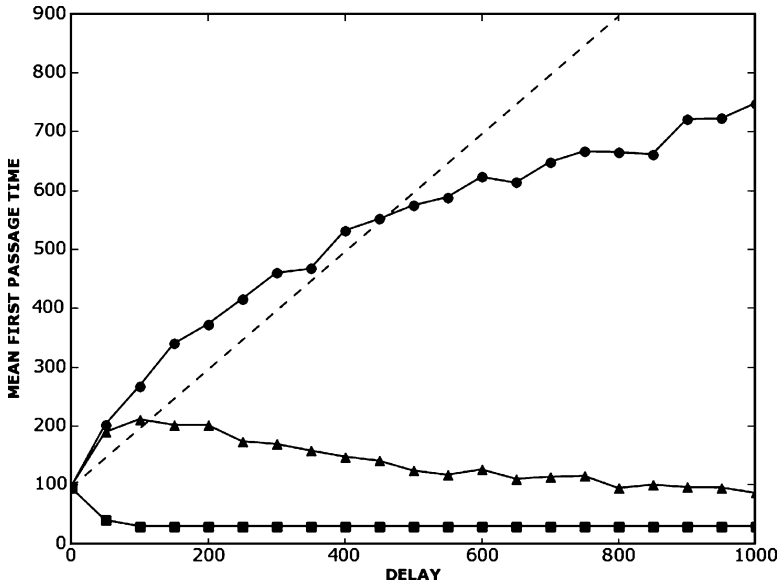
$$\frac{dx}{dt} = \begin{cases} \alpha x(t - \tau) + \eta^2 \xi(t) + K & \text{if } x(t - \tau) < -\Pi \\ \alpha x(t - \tau) + \eta^2 \xi(t) & \text{if } -\Pi \leq x(t - \tau) \leq \Pi \\ \alpha x(t - \tau) + \eta^2 \xi(t) - K & \text{if } x(t - \tau) > \Pi, \end{cases} \quad (11.97)$$

where  $\alpha$ ,  $K$ , and  $\Pi$  are positive constants. This model is a simple extension of a model proposed by Eurich and Milton [17] and states that the nervous system allows the controlled variable to drift under the effects of noise ( $\xi(t)$ ) and positive feedback ( $\alpha > 0$ ) with corrective actions (“negative feedback”) taken only when  $x(t - \tau)$  exceeds a certain threshold,  $\Pi$ . It is assumed that  $\alpha$  is small and that

$\tau < 3\alpha/2\pi$ . This means that there is one real positive eigenvalue and an infinite number of complex eigenvalue pairs whose real part is negative. The magnitude of the real positive eigenvalue decreases as  $\tau$  increases for  $0 < \tau < 3\alpha/2\pi$ . “Safety net” type controllers also arise in the design of strategies to control attractors that have shallow basins of attraction [24].

The cost to operate this discontinuous balance controller is directly proportional to the number of times it is activated. Thus the mathematical problem becomes that of determining the first passage times for a repulsive delayed random walk, i.e., the times it takes a walker starting at the origin to cross the threshold,  $\pm|I|$ . Numerical simulations of a repulsive delayed random walk indicate that the mean first passage time depends on the choice of  $\Phi_0(t)$  (see Fig. 11.9).

In principle, the most probable, or mean, first passage time can be calculated from the backward Kolmogorov equation for (11.1). We have been unable to complete this calculation. However, simulations of (11.97) indicate that this discontinuous balance controller takes advantage of two intrinsic properties of a repulsive delayed random walk: (1) the time delay which slows escape and (2) the fact that the distribution of first passage times is bimodal. Since the distribution of first passage times is bimodal, it is possible that a reset due to the activation of the negative feedback controller can lead to a transient confinement of the walker near the origin thus further slowing escape.



**Fig. 11.9** The mean first passage time,  $\hat{L}$ , for a repulsive delayed random walk as a function of  $\tau$  for three different choices of the initial function,  $\Phi_0(t)$ : (1) a constant zero function (solid line, closed circle); (2) an initial function constructed from a simple random walk with  $\tau = 0$  (solid line, closed triangle); (3) a linear decreasing initial function with end points  $x = \tau$  at  $t = -\tau$  and  $x = 0$  at  $t = 0$  (solid line, closed square). In all cases  $\Phi_0(0) = 0$ . The dashed line is equal to  $\hat{L}_{\tau=0} + \tau$ . For each choice of  $\Phi_0(t)$ , 500 realizations were calculated with  $X^* = \pm 30$  with  $d = 0.4$  and  $a = 30$

## 11.6 Concluding Remarks

In this chapter, we have shown that a properly formulated delayed random walk can provide an alternate and complimentary approach for the analysis for stochastic delay differential equations such as (11.1). By placing the study of a stochastic differential equation into the context of a random walk it is possible to draw upon the large arsenal of analytical tools previously developed for the study of random walks and apply them to the study of these complex dynamical systems. The advantages of this approach to the analysis of SDDEs include the following: (1) it avoids issues inherent in the Ito vs. Stratonovich stochastic calculus, (2) it provides insight into transient dynamics, and (3) it is, in principle, applicable to the study of delayed stochastic dynamical systems in the setting of complex potential surfaces such as those that arise in the setting of multistability. In general, the use of the delayed random walk involves solutions in the form of equations which must be solved iteratively. However, this procedure causes no practical problem given the ready availability of symbolic manipulation computer software programs such as MAPLE<sup>®</sup> and MATHEMATICA<sup>®</sup>. The use of these computer programs compliment the variety of numerical methods [3,27,29] that have been developed to investigate SDDEs, such as (11.1). However, much remains to be done, particularly the important cases in which the noise enters the dynamics in a multiplicative, or parametric, fashion [9, 10, 52]. Thus we anticipate that the subject of delayed random walks will continue to gain in importance.

**Acknowledgments** We thank Sue Ann Campbell for useful discussions. We acknowledge support from the William R. Kenan, Jr. Foundation (JM) and the National Science Foundation (Grant 0617072)(JM,TO).

## References

1. Bailey, N. T., *The Elements of Stochastic Processes*, Wiley, New York, NY, 1990.
2. Bechhoefer, J., Feedback for physicists: A tutorial essay on control, *Rev. Mod. Phys.* **77**, 2005, 783–836.
3. Bellen, A. and Zennaro, M., *Numerical Methods for Delay Differential Equations*, Oxford University Press, New York, NY, 2003.
4. Bendat, J. S. and Piersol, A. G., *Random Data: Analysis and Measurement Procedures*, 2nd Edition, Wiley, New York, NY, 1986.
5. Berg, H. C., *Random Walks in Biology, Expanded Edition*, Princeton University Press, Princeton, NJ, 1993.
6. Boukas, E-K. and Liu Z-K., *Deterministic and Stochastic Time Delay Systems*, Birkhäuser, Boston, MA, 2002.
7. Bracewell, R. N., *The Fourier Transform and its Applications*, 2nd Edition, McGraw-Hill, New York, NY, 1986.
8. Bratsun, D., Volfson, D., Tsimring, L. S. and Hasty, J., Delay-induced stochastic oscillations in gene regulation, *Proc. Natl Acad. Sci. USA* **102**, 2005, 14593–14598.

9. Cabrera, J. L. and Milton, J. G., On-off intermittency in a human balancing task, *Phys. Lett. Rev.* **89**, 2002, 158702.
10. Cabrera, J. L. and Milton, J. G., Human stick balancing: Tuning Lévy flights to improve balance control, *Chaos* **14**, 2004, 691–698.
11. Cabrera, J. L., Bormann, R., Eurich, C. W., Ohira, T. and Milton, J., State-dependent noise and human balance control, *Fluct. Noise Lett.* **4**, 2004, L107–L117.
12. Cabrera, J. L., Luciani, C., and Milton, J., Neural control on multiple time scales: Insights from human stick balancing, *Condens. Matter Phys.* **2**, 2006, 373–383.
13. Collins, J. J. and De Luca, C. J., Random walking during quiet standing, *Phys. Rev. Lett.* **73**, 1994, 907–912.
14. Davenport, W. B. and Root, W. L., *An Introduction to the Theory of Random Signals and Noise*, IEEE, New York, NY, 1987.
15. Ehrenfest, P. and Ehrenfest, T., Über zwei bekannte Einwände gegen das Boltzmannsche H-Theorem, *Phys. Zeit.* **8**, 1907, 311–314.
16. Einstein, A., Zür Theorie der Brownschen Bewegung, *Annalen der Physik* **19**, 1905, 371–381.
17. Eurich, C. W. and Milton, J. G., Noise-induced transitions in human postural sway, *Phys. Rev. E* **54**, 1996, 6681–6684.
18. Fort, J., Jana, D. and Humet, J., Multidelayed random walk: Theory and application to the neolithic transition in Europe, *Phys. Rev. E* **70**, 2004, 031913.
19. Frank, T. D., Delay Fokker-Planck equations, Novikov's theorem, and Boltzmann distributions as small delay approximations, *Phys. Rev. E* **72**, 2005, 011112.
20. Frank, T. D. and Beek, P. J., Stationary solutions of linear stochastic delay differential equations: Applications to biological systems, *Phys. Rev. E* **64**, 2001, 021917.
21. Gardiner, C. W., *Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences*, Springer, New York, NY, 1994.
22. Glass, L. and Mackey, M. C., *From Clocks to Chaos: The Rhythms of Life*, Princeton University Press, Princeton, NJ, 1988.
23. Grassia, P. S., Delay, feedback and quenching in financial markets, *Eur. Phys. J. B* **17**, 2000, 347–362.
24. Guckhenheimer, J., A robust hybrid stabilization strategy for equilibria, *IEEE Trans. Autom. Control* **40**, 1995, 321–326.
25. Guillouzic, S., L'Heureux, I. and Longtin, A., Small delay approximation of stochastic delay differential equation, *Phys. Rev. E* **59**, 1999, 3970–3982.
26. Hale, J. and Koçak, H., *Dynamics and Bifurcations*, Springer, New York, NY, 1991.
27. Hofmann, N. and Müller-Gronbach, T., A modified Milstein scheme for approximation of stochastic delay differential equation with constant time lag, *J. Comput. Appl. Math.* **197**, 2006, 89–121.
28. Hosaka, T., Ohira, T., Luciani, C., Cabrera, J. L. and Milton, J. G., Balancing with noise and delay, *Prog. Theor. Phys. Suppl.* **161**, 2006, 314–319.
29. Hu, Y., Mohammed, S.-E. A. and Yan, F., Discrete time approximations of stochastic delay equations: the Milstein scheme, *Ann. Probab.* **32**, 2004, 265–314.
30. Jenkins, G. M. and Watts, D. G., *Spectral Analysis and its Applications*, Holden-Day, San Francisco, CA, 1968.
31. Kac, M., Random walk and the theory of Brownian motion, *Am. Math. Monthly* **54**, 1947, 369–391.
32. Karlin, S. and McGregor, J., Ehrenfest urn models, *J. Appl. Prob.* **2**, 1965, 352–376.
33. Küchler, U. and Mensch, B., Langevins stochastic differential equation extended by a time-delayed term, *Stoch. Stoch. Rep.* **40**, 1992, 23–42.
34. Landry, M., Campbell, S. A., Morris, K. and Aguilar, C. O., Dynamics of an inverted pendulum with delayed feedback control, *SIAM J. Dynam. Syst.* **4**, 2005, 333–351.
35. Lasota, A. and Mackey, M. C., *Chaos, Fractals and Noise: Stochastic Aspects of Dynamics*, Springer, New York, NY, 1994.
36. Longtin, A., Noise-induced transitions at a Hopf bifurcation in a first-order delay-differential equation, *Phys. Rev. A* **44**, 1991, 4801–4813.

37. Longtin, A., Milton, J. G., Bos, J. E., and Mackey, M. C., Noise and critical behavior of the pupil light reflex at oscillation onset, *Phys. Rev. A* **41**, 1990, 6992–7005.
38. Loram, I. D. and Lakie, M., Human balancing of an inverted pendulum: position control by small, ballistic-like, throw and catch movements, *J. Physiol.* **540**, 2002, 1111–1124.
39. Loram, I. D., Maganaris, C. N. and Lakie, M., Active, non-spring-like muscle movements in human postural sway: how might paradoxical changes in muscle length be produced? *J. Physiol.* **564.1**, 2005, 281–293.
40. MacDonald, D. K. C., *Noise and Fluctuations: An Introduction*, Wiley, New York, NY, 1962.
41. MacDonald, N., *Biological Delay Systems: Linear Stability Theory*, Cambridge University Press, New York, NY, 1989.
42. Mackey, M. C. and Nechaeva, I. G., Noise and stability in differential delay equations, *J. Dynam. Diff. Eqns.* **6**, 1994, 395–426.
43. Mackey, M. C. and Nechaeva, I. G., Solution moment stability in stochastic differential delay equations, *Phys. Rev. E* **52**, 1995, 3366–3376.
44. Malkiel, B. G., *A Random Walk Down Wall Street*, W. W. Norton & Company, New York, NY, 1993.
45. Mazo, R. M., *Brownian Motion: Fluctuation, Dynamics and Applications*, Clarendon, Oxford, 2002.
46. Mergenthaler, K. and Enghert, R., Modeling the control of fixational eye movements with neurophysiological delays, *Phys. Rev. Lett.* **98**, 2007, 138104.
47. Milton, J. and Foss, J., Oscillations and multistability in delayed feedback control. In: *Case Studies in Mathematical Modeling: Ecology, Physiology, and Cell Biology* (H. G. Othmer, F. R. Adler, M. A. Lewis and J. C. Dallon (eds). Prentice Hall, Upper Saddle River, NJ, pp. 179–198, 1997.
48. Milton, J. G., Longtin, A., Beuter, A., Mackey, M. C. and Glass, L., Complex dynamics and bifurcations in neurology, *J. Theor. Biol.* **138**, 1989, 129–147.
49. Milton, J. G., Cabrera, J. L. and Ohira, T., Unstable dynamical systems: Delays, noise and control, *Europhys. Lett.* **83**, 2008, 48001.
50. Milton, J., Townsend, J. L., King, M. A. and Ohita, T., Balancing with positive feedback: The case for discontinuous control, *Philos. Trans. R. Soc.* (submitted).
51. Mohammed, S.-E. A., *Stochastic Functional Differential Equations*, Pitman, Boston, MA, 1984.
52. Mohammed, S.-E. A. and Scheutzow, M. K. R., Lyapunov exponents of linear stochastic functional differential equations. Part II. Examples and case studies, *Ann. Probab.* **25**, 1997, 1210–1240.
53. Newell, K. M., Slobounov, S. M., Slobounova, E. S. and Molenaar, P. C. M., Stochastic processes in postural center-of-pressure profiles, *Exp. Brain Res.* **113**, 1997, 158–164.
54. Niculescu, S.-I. and Gu, K., *Advances in Time-Delay Systems*, Springer, New York, NY, 2004.
55. Ohira, T., Oscillatory correlation of delayed random walks, *Phys. Rev. E* **55**, 1997, R1255–R1258.
56. Ohira, T. and Milton, J., Delayed random walks, *Phys. Rev. E* **52**, 1995, 3277–3280.
57. Ohira, T. and Sato, Y., Resonance with noise and delay, *Phys. Rev. Lett.* **82**, 1999, 2811–2815.
58. Ohira, T. and Yamane, T., Delayed stochastic systems, *Phys. Rev. E* **61**, 2000, 1247–1257.
59. Ohira, T., Sazuka, N., Marumo, K., Shimizu, T., Takayasu, M. and Takayasu, H., Predictability of currency market exchange, *Phys. A* **308**, 2002, 368–374.
60. Patanarapeelert, K., Frank, T. D., Friedrich, R., Beek, P. J. and Tang, I. M., Theoretical analysis of destabilization resonances in time-delayed stochastic second-order dynamical systems and some implications for human motor control, *Phys. Rev. E* **73**, 2006, 021901.
61. Perrin, J., *Brownian Movement and Molecular Reality*, Taylor & Francis, London, 1910.
62. Rudnick, J. and Gaspari, G., *Elements of the Random Walk*, Cambridge University Press, New York, NY, 2004.
63. Santillan, M. and Mackey, M. C., Dynamic regulation of the tryptophan operon: A modeling study and comparison with experimental data, *Proc. Natl Acad. Sci.* **98**, 2001, 1364–1369.
64. Weiss, G. H., *Aspects and Applications of the Random Walk*, North-Holland, New York, NY, 1994.

65. Wu, D. and Zhu, S., Brownian motor with time-delayed feedback, *Phys. Rev. E* **73**, 2006, 051107.
66. Yao, W., Yu, P. and Essex, C., Delayed stochastic differential equation model for quiet standing, *Phys. Rev. E* **63**, 2001, 021902.
67. Yildirim, N., Santillan, M., Horik, D. and Mackey, M. C., Dynamics and stability in a reduced model of the *lac* operon, *Chaos* **14**, 2004, 279–292.

# Index

## A

Adaptive Posicast controller, 62  
Adjoint equation, operator, 159  
Air-to-fuel ratio control, 56, 63–65, 67, 69–72, 75, 88  
Amplitude  
  stable, 198–200  
  unstable, 199, 200  
Ansatz, 224, 226  
Aperiodic dynamics, 295  
archi, 250, 254, 255  
Auto-correlation function, 306, 307, 309, 313–320, 322, 324, 329  
Automotive powertrain, 55  
Autonomous system, 96, 97, 100, 136, 138, 143, 279  
  non-autonomous system, 96, 98, 100, 103, 143–151  
Averaged coefficients, 132  
Axon, 274, 276, 299

## B

Balance control, 326, 330, 331  
Banach space, 206, 223  
Bidirectional coupling, 278  
Bilinear form, 159, 162, 171, 206, 207, 226–228, 233, 234  
Bifurcation  
  Bogdanov Takens, 225, 229  
  cyclic-fold, 135, 141  
  double Hopf, 204, 225, 229  
  diagram, 213–216, 218, 280, 282–284, 286, 287, 289–291, 294  
  existence of Hopf, 218  
  flip, 135, 141  
  inverse Hopf, 290  
  local, 157, 209

Hopf, 135, 141, 156–158, 178, 179, 182, 184, 189, 197, 203, 204, 206, 207, 209, 212–215, 217, 218, 225, 229–232, 234, 239–241, 275, 290  
Poincaré-Andronov-Hopf, 157  
  period-doubling, 135  
  secondary Hopf, 135, 141  
  subcritical, 184, 197  
Biology, 93, 98, 132, 133, 156, 203, 218, 307  
Bluetooth, 38, 42, 43, 49  
Boundary conditions  
  essential, 99  
  natural, 99  
Brownian motion, 310, 311  
Bursting, 274, 275, 277, 283, 284, 288

## C

Characteristic equation, 11–13, 96, 102, 122, 132, 158, 159, 171, 173, 179, 180, 185, 205, 216, 224, 226, 227, 231, 247  
Characteristic exponents, 96, 97  
Characteristic multipliers, 97, 98, 100–102, 125, 134, 279  
Characteristic quasipolynomial, 27, 28  
Chatter, regenerative, 94, 156  
Center (centre) eigenspace, 221–222, 226, 232, 233, 241  
Center (centre) manifold, parameter dependent, 241  
Center manifold, 158  
Center subspace, 206, 208, 209  
Chaos, 122, 123, 274  
Chay model, 301  
Chay neurons, 283–286, 288  
Chebyshev  
  collocation, 95, 109–115, 117, 125  
  polynomials, 104–111, 116, 119–124, 132

- Chebyshev-based methods, 94, 123, 124
- Collocation method, 109–111, 113, 114, 124, 250
- Complexification, 159, 170
- Conduction delay(s), 284–289
- Continuous extension, 253, 254, 256, 257
- Continuous mapping, 223
- Controller architecture, 223
- Controller area network (CAN), 223
- Control
  - data, 34, 35, 38–43
  - delayed state feedback, 116, 120–123
  - feedback, 25, 78, 305
  - Lyapunov function (CLF), 80, 132
  - optimal, 94, 123, 117–120
  - proportional-plus-integral, 65, 75
- Correlation function, 306, 307, 313–320, 322–326, 329
- Cost function, 117, 119, 120
- Coupled differential-difference equations, 1
- Coupled oscillators, 274
- Critical eigen triple, 180, 181, 183, 184
- Critical frequency, 184, 213
  
- D**
- Data sampling, 33
- dde23, 95, 250–252, 254–258, 260–264
- ddesd, 250, 251, 255–258, 264, 268, 269
- DDE.SOLVER, 263, 266–268
- Ddverk, 250, 254, 255
- Delay differential equations (DDEs), 1–28, 31, 69, 93–97, 102, 103, 109, 110, 123–125, 131–133, 138, 151, 152, 156–162, 164, 167, 173, 186, 203, 206, 221, 222, 225, 241, 245–269, 305, 309, 332
- DDE-Biftool, 213–218
- Delays
  - constant, 59, 60, 132, 133, 138, 151, 152, 246, 248, 251, 254
  - global, 34
  - incommensurate, 27, 28
  - multiple, 4, 69, 249–251
  - periodic, 132–135, 151
  - varying, 32, 132–135, 151, 259
- Delay differential equations, time periodic, 69
- Delayed Fokker-Planck equation, 321–322
- Delayed Langevin equation, 306, 322, 324
- Delayed random walk (DRW), 305–332
- Dendritic, 274
- Diesel engine, 56, 64, 78–80, 82, 87, 88
- Differentiable functional, 8
- Differential-difference equations, 1, 13–16
- Discontinuity tree, 250, 254
- Discretization methods, semi-discretization, 151
- Discretized Lyapunov Functional Method, 23–27
  
- E**
- Economics, 93, 307
- Ehrenfest model, 306, 307
- Eigenfunctions, 125, 206, 207, 209, 232
- Eigenspace, 162, 165, 171, 173, 221, 222, 224–226, 232, 233, 235, 241
- Eigenvalues
  - complex, 159, 162, 170
  - critical, 162, 166, 171, 176, 179, 207
  - zero, 225
- El-Niño Southern Oscillation (ENSO), 245, 247, 252, 259–260, 263
- Engines
  - diesel, 56, 64, 88
  - gasoline, 56, 57, 64, 88
  - lean-burn, 64
- Enhanced neural synchrony, 298
- Ensemble average, 308
- Error estimation, 255
- Euclidean norm, induced norm for matrices, 223
- Event location, 254, 255–258, 260
- Exhaust gas recirculation (EGR), 56, 78–83, 86–88
- Exponential
  - polynomials, 171, 180
  - stabilization, 32, 45
  
- F**
- Fast-spiking (FS) model, 289–291, 302
- Feedback, 3, 4, 23, 25, 31, 35, 48, 61, 64, 67, 71–73, 76, 78, 80–82, 85, 94, 103, 116, 120–123, 141, 155, 203, 245, 259, 275, 305, 306, 316, 326, 330–331
- Finite dimensional approximation, 95, 108, 135, 138
- Finite element methods, 93–126, 132
- Finite horizon, 117
- Finite spectrum assignment, 61, 84
- Firing pattern, 273, 302
- FitzHugh-Nagumo (FHN) model, 274, 275
- Floquet
  - multipliers, 95, 97, 110, 112, 134, 135, 141, 279
  - theory, 95, 103, 132–134, 151, 189, 279
  - transition matrix, 95–97, 132, 143
- Fokker-Planck equation, 306, 307, 312–313, 317, 321–322
- Fortran 90/95, 251, 257, 258

Fourier components, 295  
 Fourier transform, 310, 314, 315, 320, 324  
 Frequency-domain approach, 27  
 Frequency entrainment, 277, 280  
 Full discretization, 132, 133  
 Functional differential equation (FDE), 223, 224, 227, 231, 232

## G

Gain matrix, 80, 121–123  
 Gain scheduling, 32, 45–51, 69–71, 88  
 Geometry turbocharging, 56, 88  
 Gramm-Schmidt orthogonalization, 163  
 Group of operators, 160

## H

Harmonic balance, 203, 215  
 Heated exhaust gas oxygen (HEGO), 63, 65, 66, 75–78  
 Hilbert space, 170  
 Hindmarsh-Rose (HR) model, 274  
 Hopf bifurcation, 135, 141, 156–158, 178, 179, 182, 184, 189, 197, 203, 204, 206, 207, 209, 212–218, 225, 229–232, 234, 239–241, 275, 290  
 Hopf condition, 172, 184  
 Hodgkin-Huxley (HH) model, 274, 275, 299–301

## I

Idle speed control (ISC), 56–63, 65, 67, 88  
 Infinitesimal generator, 169, 170, 173, 174, 178, 179, 185  
 Initial condition, 5–7, 10, 13, 117, 134, 165, 167, 168, 189, 223, 256, 307, 329  
 Initial function, 35, 95, 107, 111, 134, 167, 218, 223, 306, 331  
 Initial value problems (IVPs), 247–249, 262  
 Inner product, 159, 161–163, 165–166, 170–171, 179  
 Invariant manifold, 158, 186

## J

Jacobian, 83, 88, 223, 278

## K

Kronecker product, 107

## L

Lagrange identity, 161, 170, 171  
 Langevin equation, 306, 317, 320, 322, 324–325  
 Laplace transform, 84  
 Legendre polynomials, 99, 124, 125

Liénard equation, 203–218  
 Linstedt-Poincaré method, 203, 215  
 Limit cycle, 76–78, 156, 197, 215, 217, 218, 241, 276, 277, 279  
 Linear matrix inequalities (LMI), 24, 27, 36, 37, 47, 48, 51, 56, 61, 72, 73, 89  
 Linearization, 71, 73, 83, 85, 88, 110, 221, 224–226  
 Linear space, 163, 164  
 Lipschitz, 6, 223  
 LMI Toolbox, 24  
 Local extrapolation, 255  
 Lossless transmission line, 2  
 Lyapunov exponents, 279, 280, 283, 284  
 Lyapunov functional, 23, 24, 27, 132–134  
 Lyapunov-Kraskovskii functional, quadratic, 4, 13–16, 27

## M

Machine tool vibrations, 94, 203  
 Map, 95, 96, 98, 100, 102, 108, 117, 121, 137–140, 142  
 Mathieu equation  
   damped, 102, 138, 142–144  
   delayed, 102, 109, 119–120, 122–123, 138, 142–144  
 MAPLE, 230, 231, 232, 241, 332  
 MATLAB, 24, 95, 112, 125, 213, 251, 252, 257, 258, 260, 262, 263, 268, 269  
 MATHEMATICA, 121, 332  
 Markov chain, 62  
 Master, 31–51  
 Master-slave structure, 33, 38  
 Matrices  
   coefficient, 11, 13, 108, 132, 135, 143, 146, 149, 157  
   periodic coefficient, 135, 146  
   positive definite, 5, 12, 37, 48, 117  
   negative definite, 5  
 Matrix  
   block diagonal, 225, 226  
   characteristic, 231, 232  
 Maximum Brake Torque, 57, 58  
 Method of multiple scales, 203  
 Method of steps, 69, 248, 252, 253  
 Miabot, 38, 41, 42, 43, 49  
 Milling  
   constant spindle speed, 133, 138, 144–151  
   downmilling, 147  
   process, 113, 115, 123, 133, 144–151  
   variable spindle speed, 133, 138, 148–151  
 Monodromy matrix, pseudo, 147

## Monodromy operator

- finite-dimensional approximation, 138
- infinite dimensional, 95, 97, 132, 135

Morris-Lecar (ML) model, 300–301

Multistability, 274, 275, 292–297, 332

## N

Neural networks, 131–134, 203, 246, 268, 273–275, 297

Neuronal system, 273–302

Neurons

- firing, 273–302

- synchronization, 273–302

Networked control system

- Ethernet, 32

- Internet, 32, 33, 39, 42, 43, 45

Network time protocol (NTP), 33, 38

Noise, 75, 275, 297, 305–332

Nonhyperbolic, equilibrium point, 221, 222, 241

Nonlinear functional, 223, 230

Normal form, 158, 159, 161, 176, 186, 188–189, 194–195, 229, 241

Numerical methods, 69, 109, 132, 151, 250–258, 269, 332

## O

Observer

- error dynamics, 82, 83, 85

- design, 35–36, 62, 78–88

Ode23, 251, 252

Operator differential equation, 206–209

Ordinary differential equations (ODEs), 94–97, 105, 111, 210, 221, 222, 224, 225, 228, 229, 238, 239, 245–258, 269

Orthogonal decomposition, 162–164, 166–167, 172–176, 185–186

## P

Packet losses, 33

Padé approximation, 60, 61

Parseval's formula, 314

Partial derivatives, 187, 192

Partial differential equations, 2, 228, 330

Phase locking, 277, 279

Physiology, 132, 156, 275, 305

Poincaré-Lyapunov constant, 157, 206, 212, 217, 218

Poincaré-Lyapunov formula, 217

Poincaré section, 279

Population dynamics, 156, 203

Power series, 104, 228

Postural sway, 306, 307, 326–330

## Probability

- density, 306, 307, 311, 316

- distribution function, 309–311, 316, 318

Propagated discontinuities, 254, 255

Proportional-plus-Integral control, 65, 75

## Q

Quadratic potential, 306, 316–318, 320, 321, 323, 324, 328

Quasipolynomial, 27, 28

Quasiperiodic, 132, 149, 274

## R

Radar5, 250, 255, 256

Random walk

- attractive, 318, 322, 324

- repulsive, 318, 331

Regenerative chatter, 94, 156

Residual, 99, 101, 103, 255, 256, 264

Riesz representation theorem, 5

Robotics, 25, 93, 98

Rocking suitcase, 260–268

Routh-Hurwitz criterion, 121

Runge-Kutta, 250, 252–255

## S

Semi-discretization, 95, 123

Semigroup of operators, strongly continuous, 168, 170, 179

Singular, 5, 102, 241, 246, 254

Singular problems, 246

Slave, 31–51

Solution history queue, 257

Solution operator, 134, 161, 167, 168

Solution smoothing, 249

Spark reserve, 57, 61

Spatiotemporal dynamics, 276, 292–297

Spectral observability, 84

Stability

- asymptotic, 7–10, 16, 21, 23, 89, 96–98, 100, 108, 121, 122, 125, 132, 224, 278, 279

- boundaries, 70, 100, 112, 125, 140, 147, 181, 182

- charts, 69, 94, 95, 103, 110, 114, 115, 123, 139, 140, 142, 143, 147–150, 182

- global, 7, 8, 10, 35, 37

- input-to-state, uniformly, 7, 8, 11, 16, 23

- limit, 181, 182, 185

- linear, 4, 10, 204–206, 217

- periodic solution, 110, 132–135, 158

- robust, 89–90

- stochastic, 71–73, 89–90

- trivial solution, 5, 7, 8, 138, 224, 226, 279
- uniform, 7–11, 16, 23, 45–48, 51
- Stochastic delay differential equation (SDDE), 305–307, 309, 332
- Stochastic dynamical system, 316, 329, 332
- Stochastic resonance, 306
- Sunflower equation, 214–218
- Symbolic algebra, 222, 240, 241
- Symbolic computation, 221
- Synaptic, 274–276, 283, 284, 289, 290, 294, 297, 302
- Synchronization
  - clock, 33, 38–39, 51
  - manifold, 278
  - neuron systems, 274–276
  - phase, 275, 279, 280, 284, 293, 294
- System identification, 72
- T**
- Temporal finite element analysis (TFEA), 97, 98, 101–102, 114–116, 123–126
- Time-delay systems
  - neutral, 26
  - retarded, 23, 24
- Time-domain approach, 84, 133
- Time lag, 131, 274, 275, 281, 297
- Time-varying delay, approximation, 136, 137
- Three-way-catalyst (TWC), 63
- Time finite element analysis (TFEA), 98, 101–102, 114–116, 123–125
- Transmission delay, 32–34, 39, 43, 274
- Trial functions, 98–100, 102, 124, 125
- Turing-Hopf instability, 295
- U**
- Uncertainty, 72–74, 76
- Uniformly non-atomic function, 6
- Universal Exhaust gas oxygen (UEGO) sensor, 63, 65
- V**
- Vanishing delays, 246, 250
- Variable geometry turbine (VGT), 78, 79, 81, 83, 86, 87
- Variance, 214, 307, 312, 314, 318–320, 323, 325, 328, 330
- Variation of parameters, 116–117, 124
- Virtual delay, 32
- W**
- Waves
  - traveling, 275, 295
  - stationary, 274, 294, 295
- Wiener-Khintchine theorem, 307, 314, 316, 319, 324, 325
- Weiner process, 320, 324