IntechOpen

Nonlinear Systems

Design, Analysis, Estimation and Control

Edited by Dongbin Lee,
Tim Burg and Christos Volos

NONLINEAR SYSTEMS - DESIGN, ANALYSIS, ESTIMATION AND CONTROL

Edited by **Dongbin Lee, Tim Burg** and **Christos Volos**

Nonlinear Systems - Design, Analysis, Estimation and Control
http://dx.doi.org/10.5772/61494
Edited by Dongbin Lee, Tim Burg and Christos Volos

Contributors

Majdi Mansouri, Marwa Chaabane, Hazem Nounou, Ahmed Ben Hamida, Imen Baklouti, Nouha Jaouab, Mohamed N. Nounou, Youmin Tang, Sie Long Kek, Kok Lay Teo, Mohd Ismail Abd Aziz, Maria Vassileva, Alicia Cordero, Juan R. Torregrosa, Jun Yoneyama, Kenta Hoshino, Daisuke Sonoda, Anh Tuan Le, Soon-Geul Lee, Jenq-Lang Wu, Chee-Fai Yung, Tsu-Tian Lee, Issa Tall, Sandile Motsa, Mutua Samuel, Stanford Shateyi, Rafael Morales, Lidia M. Belmonte, Antonio Fernández-Caballero, J.A. Somolinos, Misir Mardanov, Jeong Ryeol Choi, Salah Menouar, Hector E. Nistazakis, Christos Volos, Ioannis Kyprianidis, Ioannis Stouboulos, George S. Tombras, Kuan Min Wang, Yuan-Ming Lee

Notice

First published in Croatia, 2016 by INTECH d.o.o.
eBook (PDF) Published by IN TECH d.o.o.
Place and year of publication of eBook (PDF): Rijeka, 2019.
IntechOpen is the global imprint of IN TECH d.o.o.
Printed in Croatia

Legal deposit, Croatia: National and University Library in Zagreb

Additional hard and PDF copies can be obtained from orders@intechopen.com

Nonlinear Systems - Design, Analysis, Estimation and Control
Edited by Dongbin Lee, Tim Burg and Christos Volos

p. cm.

Print ISBN 978-953-51-2714-7

Online ISBN 978-953-51-2715-4

eBook (PDF) ISBN 978-953-51-4173-0

Meet the editors

Dr. Dongbin Lee is with the faculty in MMET Department at OIT where he has taught several courses—robotics, automation, instrumentation, control, circuits, power systems, and programming—and is advising undergraduate research projects while teaching robotics for graduate students. He is the director of the robotics lab and has served as the adviser to a student club or robotics team. He is serving a couple of editorial boards such as in Frontiers journals and IJARS in In-Tech. Dr. Lee has two main research areas: nonlinear controls and unmanned vehicle systems, where he served as the webmaster officer in Keystone AUVSI Chapter. He worked at the Villanova University as a research associate right after he received a PhD degree at Clemson University, SC, USA, focusing on controls and robotics including UAS, UUV, and haptics.

Dr. Tim Burg is a professor in the Department of Veterinary Biosciences and Diagnostic Imaging in the College of Veterinary Medicine since 2016 and the director of STEM Education at the University of Georgia, GA, after working at Kansas State University for a year and a half. He was a faculty in the Department of Electrical and Computer Engineering and an investigator in the Institute for Biological Interfaces of Engineering (IBIOE) with the Department of Biological Engineering at Clemson University. He served as a chairman of the Scientific Research Society Sigma Xi until 2009 as its vice president. He has published a book Nonlinear Control of Electric Machinery and extensively in prestigious journals and has made numerous presentations at national and international conferences. Dr. Burg has also served on the program committee of various top-ranked international conferences and workshops and has refereed for the IEEE society in addition to many journals and conferences.

Dr. Christos Volos received his Physics Diploma, his MSc in Electronics, and his PhD in Chaotic Electronics in 1999, 2002, and 2008, respectively, all from the Aristotle University of Thessaloniki. He worked from 2004 to 2008 as a scientific associate at the Technological Educational Institutes of Thessaloniki and Serres. He currently serves as an assistant professor at the Physics Department of the Aristotle University of Thessaloniki. He is a member of the research lab of "nonlinear circuits—systems and complexity" of the Physics Department of the Aristotle University of Thessaloniki. Dr. Volos has more than 120 publications as author or coauthor in international journals, books, international and national conferences. He participated as reviewer or member of international program committees in 30 international conferences and he is the reviewer in more than 30 international journals. Finally, he is a member of the following scientific societies: Hellenic Physical Society (member since 1999) and Hellenic Radio-electronics Society (member since 2003).

Contents

Preface

This book is a research monograph to detail recent developments of nonlinear systems and control. The book selected a collection of nonlinear systems, which range from quantum mechanics, chaotic systems to nonlinear dynamical systems such as unmanned vehicle platform and 3D crane system. More interestingly, the book covers not just scientific and engineering problems but also earth data assimilation and economic development modeling, that is, from theories to applications in real-world issues. This book has a couple of analytic tools for nonlinear control problems, such as feedback linearization, partial differential equation, Frobenius theorem, Lyapunov theory and its exponents, Nikiforov-Uvarov method, eigenvalue/eigenvector, pseudocomposition (predictor-corrector), or stochastic methods including Kalman filtering. Many practical systems or natural phenomena are nonlinear, so from a scientist or engineer's perspective, we interpret the system and represent the system using scientific approaches, including physics-based analysis and chemical or social analysis, where the tools are basically mathematics. Then, they interpret the system characteristics and try to represent the system using scientific approaches to get comprehensive results of such nonlinear or complex systems. If the scientific structure and analysis are sound, the system or phenomenon may have unique solutions, or it would not be a surprise if no solution exists nor any tools or solutions. During the last couple of decades, serious efforts using estimation and many control systems with those analysis tools have helped to solve many complex nonlinear systems. Here we would like to narrow down the contents to some nonlinear dynamical systems and their controls.

Solving nonlinear system is a daunting challenge, while analyzing nonlinear system is not easy because no universal solution is available, but the authors in this book have demonstrated what would be the system responses and behaviors in their fields, what would be the appropriate analysis tools, or how to get reasonable results in terms of convergence (region of attraction)—those are the kinds of concerns they want to solve first. As the system goes more complicated or mixed, more analysis tools are needed to understand the system due to its mutual interaction or independent action. Furthermore, variables are getting bigger, sophisticated, or specialized, and sometimes, only one of analytic tools is not sufficient to describe. Thanks to the development of hardware/software of electronics with faster computing technology, detailed analysis tools have been helping researchers or scientists to get more data or be able to solve higher-order, multidimensional, or more complex nonlinear control problems and real-world issues. Thus, nonlinear analysis tools, simulation, or experimental methods got to be advanced, extended to a wide and deep as well as quantitatively digitalized measurement system that helps to acquire more useful (visualized) data due to some phenomenal researchers. Hence, many nonlinear systems are resolved alongside the development of tools and control methods. On the other hand, linear system analysis and a

set of tools such as linearization or partial differential equations are also used to analyze many nonlinear systems in a piecewise linear manner.

The book consists mainly of two parts as follows: the first section includes design, analysis, methods, and techniques of nonlinear system and the second section includes controls and applications of nonlinear systems. The following are brief outlines of each chapter.

In the first section, "Nonlinear Systems: Design, Analysis, and Estimation Methods," the authors in Chapter 1 present that quasilinearization technique can be simplified to partial differential equation (PDE) while decomposing time domain into smaller subintervals by applying spectral allocation method and extending up to boundaries with continuity condition. In Chapter 2, the authors suggest an algorithm to find the change of coordinates and feedback for partial feedback linearization that are useful tools to convert nonlinear control system into partial linearization with feedback if the system has full rank and is involutive by Frobenius theorem. The authors in Chapter 3 investigate quantum characteristics of singular potential of a particle using Nikiforov-Uvarov method and solved analytically the full wave functions with the evaluation of eigenfunctions with eigenvalues. Chapters 4 to 7 deal with estimation methods using Kalman filtering based on stochastic approach while solving nonlinear systems using iterative approach. The authors in Chapter 4 provide discrete-time nonlinear stochastic control problem solving iteratively using model-based optimal control by adding adjustable parameters to a continuous stirred-tank reactor model. In Chapter 5, the authors compare classical methods with multipoint iterative approaches such as composition of known methods, weight function procedure, and pseudocomposition to solve nonlinear systems. A couple of different iterative Kalman filtering algorithms with simulation examples is provided with two comparative studies in terms of state accuracies, estimation errors, and convergence where ISRCDKF provides the most improved state accuracies than the other techniques. A very thoughtful review of ensemble-based estimation methods is present in Chapter 7 where the authors provided many analyses, derivations, and discussions of Kalman filtering and particle filtering approaches. More importantly, the authors put more weights on those solutions to high dimensional systems in earth sciences where the novelty of this chapter lies in.

In the second section, "Control and Applications of Nonlinear Dynamical Systems," we have selected a couple of control approaches such as optimal, nonlinear, and output feedback. These methods with analytic tools are applied to nonlinear dynamical systems to solve practical nonlinear control problems. In Chapter 8, the authors deal with optimal control problem with retarded control of singular situation in which they first optimize the conditions and obtain necessary conditions, design optimal solution, and then apply the controller to Legendre equation to demonstrate the results. The authors in Chapter 9 present simultaneous H-∞ control for nonlinear system under strict-feedback form, which is more challenging, but they used backstepping approach based on systematically control storage functions (CSF). Chapter 10 provides a control approach over three-dimensional overhead crane system using two separated subsystems, actuated and unactuated, in which the actuated subsystem is used to linearize nonlinear feedback states, whereas the unactuated subsystem is combined with the linear system. In Chapter 11, the authors develop dynamic model of twin-rotor helicopter, manufactured by Feedback Instruments Inc., which is a nonlinear cascaded, coupled structure. They also developed its control algorithm, which carries out two electrical and mechanical parts and provided numerical results. In Chapter 12, the authors design a nonlinear controller to target antisynchronization/synchronization states

based on Birkhoff-Shaw nonautonomous chaotic coupled systems, and the stability of the systems is ensured by Lyapunov exponent theorem. With electronic circuit that models the coupling scheme, the authors are to verify the feasibility of their proposed design. Numerical methods based on linear matrix inequalities (LMI) provide solutions to stabilization of nonlinear control problems in Chapter 13. With the design of output feedback laws based on the sum-of-squares (SoS) decomposition with state-dependent LMI, nonlinear polynomial systems can be stabilized via generalization as well as provide suitable analysis for stability through Lyapunov analysis. In the final chapter of this book, the authors try to analyze a practical nonlinear modeling problem. The authors examine thoroughly whether the stock return could be a leading indicator of economic growth in the depression period. In order to analyze nonlinear phenomena between economic development and stock return, a nonlinear dynamic data model is constructed with new current depth of recession indicator, especially fluctuations of stock returns, which are highly correlated with economic activities. They propose that the stock return can considerably explain the economic growth in the recession period according to the country's development level and business cycle stages.

Hence, this book is the culmination of their research and efforts. I hope it will be a good reference for the researchers and that it could provide good insights to obtain the solution for practical problems. Also it is an honor to edit these phenomenal papers. Special thanks go to InTechOpen for the opportunity and Edi Lipović, the Publishing Process Manager.

Dongbin Lee, Ph.D.
Assistant Professor/Director of Oregon Tech Robotics Lab,
Adviser to Oregon Tech Unmanned Systems Club
MMET Dept, Oregon Institute of Technology (Oregon Tech),
Klamath Falls, Oregon, United States of America

Section 1

Nonlinear Systems: Design, Analysis, and Estimation Methods

Chapter 1

Solving Nonlinear Parabolic Partial Differential Equations Using Multidomain Bivariate Spectral Collocation Method

Motsa Sandile Sydney, Samuel Felix Mutua and Shateyi Stanford

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64600

Abstract

In this study we introduce the multidomain bivariate spectral collocation method for solving nonlinear parabolic partial differential equations (PDEs) that are defined over large time intervals. The main idea is to reduce the size of the computational domain at each subinterval to ensure that very accurate results are obtained within shorter computational time when the spectral collocation method is applied. The proposed method is based on applying the quasi-linearization technique to simplify the nonlinear partial differential equation (PDE) first. The time domain is decomposed into smaller nonoverlapping subintervals. Discretization is then performed on both time and space variables using spectral collocation. The approximate solution of the PDE is obtained by solving the resulting linear matrix system at each subinterval independently. When the solution in the first subinterval has been computed, the continuity condition is used to obtain the initial guess in subsequent subintervals. The solutions at different subintervals are matched together along a common boundary. The examples chosen for numerical experimentation include the Burger's-Fisher equation, the Fitzhugh-Nagumo equation and the Burger's-Huxley equation. To demonstrate the accuracy and the effectiveness of the proposed method, the computational time and the error analysis of the chosen illustrative examples are presented in the tables.

Keywords: bivariate interpolation, spectral collocation, quasi-linearisation, multi-domain approach, non-linear evolution PDEs

1. Introduction

Most practical problems which model systems in nature lead to nonlinear partial differential equations (PDEs). This is evident in the fields of chemistry, physics, biology, mathematics and engineering. Many assumptions have been made to make some nonlinear PDEs solvable. It has been reported that a vast number of nonlinear PDEs that are encountered in these fields are difficult to solve analytically [1]. The investigation of solutions of such nonlinear PDEs has then been of key interest to many researchers due to their potential applications and more effort has been devoted to search for better and more efficient solution methods for these nonlinear models [2, 3].

The nonlinear PDEs that are solved in this study include the generalized Burger's-Fisher equation, the generalized Burger's-Huxley equation and the Fitzhugh-Nagumo equation. The generalized Burger's-Fisher equation appears in many applications such as shock wave formation, fluid mechanics, turbulence, traffic flows, gas dynamics, heat conduction and sound waves via viscous medium among other fields of applied science [4–6]. The generalized Burger's-Huxley equation models the interaction between reaction mechanisms, diffusion transports and convection effects [7–11]. The Fitzhugh-Nagumo equation arises in genetics, biology, and heat and mass transfer [12, 13].

A number of methods have been applied to solve the nonlinear PDEs such as spectral collocation method [7, 8], Adomian decomposition method [9], homotopy perturbation method [14] and the variational iteration method [4]. The spectral methods have been reported to be strikingly successful if the problem has a smooth solution and falls into various categories, namely Galerkin, Tau and collocation-based methods [15], and therefore, recent advances in the development of numerical methods for solving nonlinear PDEs has focused spectral-based approaches as they require a few grid points to give very accurate results and take less computation time. The spectral collocation-based methods are used often, chiefly because they offer the simplest treatment of boundary conditions. A newly developed spectral collocation method for solving nonlinear PDEs is the bivariate spectral quasi-linearization method (QLM) [16]. This method approximates the solution of the PDE using a bivariate Lagrange interpolation polynomial [17]. It applies quasi-linearization method of Bellman and Kalaba [18] to simplify the nonlinear PDE which is then discretized using spectral collocation on both time and space variables. The method has successfully been used to solve problems defined over shorter time intervals [16]. However, it has been observed that when this method is applied to solve problems defined over large-time intervals, there is no guarantee that the resulting approximate solution will be accurate [16].

In this study, we describe the multidomain bivariate spectral collocation method (MDBSCM) to solutions of nonlinear parabolic PDEs defined over large-time intervals. The MDBSCM is based on decomposing the given domain of approximation in the time variable into smaller subintervals and then solving the PDE independently in each subinterval using the bivariate spectral collocation method. The multidomain approach has been applied to solve nonlinear ordinary differential equations that model chaotic systems described as 1st order systems of equations [19–21]. In this study the same idea is extended to solutions of nonlinear parabolic

http://dx.doi.org/10.5772/64600

PDEs. In the description of the method, the algorithm is kept as simple as possible, while retaining the heart of generality to cover many applications. The extent of the discussion of multidomain approach in this study is limited to nonoverlapping subintervals only.

2. Method of solution

In this section, we describe the algorithm to describe how the multidomain bivariate spectral collocation method can be applied to solve nonlinear parabolic PDEs. We shall consider a general second-order nonlinear PDE,

$$\frac{\partial u}{\partial t} = F\left(u, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right), \quad x \in (a,b), t \in [0,T], \tag{1}$$

subject to boundary conditions

$$u(a,t) = g_a(t), \quad u(b,t) = g_b(t), \tag{2}$$

and initial condition

$$u(x,0) = f(x), \tag{3}$$

where $u(x, t)$ is the required solution, $f(x)$, $g_a(t)$ and $g_b(t)$ are known functions and F is a nonlinear operator operating on u and its first and second spatial derivatives.

2.1. The quasi-linearization method

The quasi-linearization method (QLM) of Bellman and Kalaba [18] is a technique that is used to simplify nonlinear ordinary and partial differential equations. The technique has been adopted and generalized in further studies presented in [22, 23]. The QLM is based on the Newton-Raphson method and is constructed from the linear terms of Taylor series expansion about an initial approximation to solution. The QLM assumes that the difference between solutions at two successive iterations denoted by u_s and u_{s+1} is very small. Applying the QLM on Eq. (1) yields

$$\begin{aligned} F(u,u',u'') \approx{}& F(u_s,u_s',u_s'') + \frac{\partial F}{\partial u}(u_s,u_s',u_s'')(u-u_s) + \frac{\partial F}{\partial u'}(u_s,u_s',u_s'')(u'-u_s') \\ &+ \frac{\partial F}{\partial u''}(u_s,u_s',u_s'')(u''-u_s''), \end{aligned} \tag{4}$$

where prime denotes differentiation with respect to x and s denotes the iteration level. Eq. (4) can be written in compact form as

$$F(u,u',u'') \approx F(u_s,u'_s,u''_s) + \sum_{\gamma=0}^{2} \frac{\partial F}{\partial u^{(\gamma)}}(u_s,u'_s,u''_s)\left(u^{(\gamma)} - u_s^{(\gamma)}\right), \tag{5}$$

where $u^{(0)} = u$. Using the expanded form of Eq. (5) in Eq. (1), we obtain the QLM scheme for approximating the solution $u_{s+1}(x, t)$ at the $(s + 1)$th iteration level as

$$\alpha_{2,s}(x,t)u''_{s+1} + \alpha_{1,s}(x,t)u'_{s+1} + \alpha_{0,s}(x,t)u_{s+1} - \dot{u}_{s+1} = R_s(x,t), \tag{6}$$

where

$$\begin{aligned} \alpha_{\gamma,s}(x,t) &= \frac{\partial F}{\partial u_s^{(\gamma)}}(u_s,u'_s,u''_s), \quad \gamma = 0,1,2, \\ R_s(x,t) &= \sum_{\gamma=0}^{2} \alpha_{\gamma,s}(x,t)u_s^{(\gamma)} - F(u_s,u'_s,u''_s). \end{aligned} \tag{7}$$

The dot here denotes differentiation with respect to the time t. Starting with an initial approximation u_0, the QLM scheme is solved iteratively until a solution with desired accuracy requirements is obtained. The multidomain approach is implemented on the linearized scheme (6) as illustrated below. For the purpose of this study, we shall apply the multidomain approach on the time (t) variable only.

Let $t \in \Gamma$ where $\Gamma \in [0, T]$. The domain Γ is decomposed into p nonoverlapping intervals as

$$\Gamma_k = [t_{k-1}, t_k], \quad t_{k-1} < t_k, \quad t_0 = 0, \quad t_p = T, \quad k = 1,2,\ldots,p. \tag{8}$$

The domain $t \in [t_{k-1}, t_k]$ in each of the kth subdomain is first transformed to $\tau \in [-1, 1]$ using the linear transformation

$$t = \frac{1}{2}\left(t_k - t_{k-1}\right)\tau + \frac{1}{2}\left(t_k + t_{k-1}\right), \tag{9}$$

before the spectral collocation is applied. Similarly, the spatial domain $x \in [a, b]$ is transformed to $\eta \in [-1, 1]$ using the linear transformation

$$x = \frac{1}{2}(b-a)\eta + \frac{1}{2}(b+a). \tag{10}$$

http://dx.doi.org/10.5772/64600

The collocation nodes are the symmetrically distributed Gauss-Lobatto grid points defined on the interval [−1, 1] by,

$$\left\{\tau_j\right\}_{j=0}^{M} = \cos\left(\frac{j\pi}{M}\right), \quad \left\{\eta_i\right\}_{i=0}^{N} = \cos\left(\frac{i\pi}{N}\right). \tag{11}$$

To distinguish between the solutions at different subdomains we shall use, $u^{(k)}$, $k = 1, 2, \ldots, p$, to denote solution at the kth subinterval. The PDE is solved independently in each subinterval. In the first subinterval we must solve,

$$\alpha_{2,s}(x,t)\frac{\partial^2 u_{s+1}^{(1)}}{\partial x^2} + \alpha_{1,s}(x,t)\frac{\partial u_{s+1}^{(1)}}{\partial x} + \alpha_{0,s}(x,t)u_{s+1}^{(1)} - \frac{\partial u_{s+1}^{(1)}}{\partial t} = R_s^{(k)}(x,t),\ x \in [a,b],\ t \in [0,t_1], \tag{12}$$

subject to boundary and initial conditions

$$u^{(1)}(a,t) = g_a(t), \quad u^{(1)}(b,t) = g_b(t), \quad u^{(1)}(x,0) = f(x). \tag{13}$$

After the solution in the first interval Γ_1 has been computed, the solutions at the subsequent kth subinterval are computed by using the solution at the right hand boundary of the $(k-1)th$ interval as an initial solution. Thus in the next subintervals, $k = 2, 3, \ldots, p$, we must solve

$$\begin{aligned} &\alpha_{2,s}(x,t)\frac{\partial^2 u_{s+1}^{(k)}}{\partial x^2} + \alpha_{1,s}(x,t)\frac{\partial u_{s+1}^{(k)}}{\partial x} + \alpha_{0,s}(x,t)u_{s+1}^{(k)} - \frac{\partial u_{s+1}^{(k)}}{\partial t} = R_s^{(k)}(x,t), \\ &x \in [a,b],\ t \in [t_{k-1},t_k], \end{aligned} \tag{14}$$

subject to boundary and initial conditions

$$u^{(k)}(a,t) = g_a(t), \quad u^{(k)}(b,t) = g_b(t), \quad u^{(k)}(x,t_{k-1}) = u^{(k-1)}(x,t_{k-1}). \tag{15}$$

In the solution process, the approximate solution that is searched for takes a form of a bivariate Lagrange interpolation polynomial. The solution at each subinterval is approximated as

$$u^{(k)}(x,t) \approx U^{(k)}(\eta,\tau) = \sum_{p=0}^{N}\sum_{q=0}^{M} U^{(k)}(\eta_p,\tau_q)L_p(\eta)L_q(\tau). \tag{16}$$

The first and second spatial derivatives are evaluated at the collocation nodes (η_i, τ_j) for $j = 0, 1, 2, \ldots, M$ as follows

$$\frac{\partial u^{(k)}}{\partial x}(\eta_i,\tau_j)=\mathbf{D}\mathbf{U}_j^{(k)}=\left(\frac{2}{b-a}\right)\hat{\mathbf{D}}\mathbf{U}_j^{(k)},\quad \frac{\partial^2 u^{(k)}}{\partial x^2}(\eta_i,\tau_j)=\mathbf{D}^2\mathbf{U}_j^{(k)} \tag{17}$$

where $\widehat{\mathbf{D}}=\left(\frac{b-a}{2}\right)\mathbf{D}$ of size $(N+1)\times(N+1)$ is the standard first-order Chebyshev differentiation matrix as defined in [15]. The time derivative is evaluated at the collocation nodes (η_i, τ_j) for $i = 0, 1, 2, \ldots, N$ as

$$\frac{\partial u^{(k)}}{\partial t}(\eta_i,\tau_j)=\sum_{q=0}^{M}d_{j,q}\mathbf{U}_q^{(k)}=\sum_{q=0}^{M}\left(\frac{2}{t_k-t_{k-1}}\right)\hat{d}_{j,q}\mathbf{U}_q^{(k)}, \tag{18}$$

where $\widehat{d}_{j,q}=\left(\frac{t_k-t_{k-1}}{2}\right)d_{j,q}$, $j, q = 0, 1, 2, \ldots, M$ of size $(M + 1) \times (M + 1)$ is the standard first-order Chebyshev differentiation matrix,

$$\mathbf{U}_j^{(k)}=[u^{(k)}(x_0,t_j),u^{(k)}(x_1,t_j),\ldots,u^{(k)}(x_N,t_j)]^T \tag{19}$$

and T denotes matrix transpose. Using the definitions (17)–(18), we express Eq. (6) in matrix form as

$$\left[\alpha_{2,s}(\mathbf{x},t_j)\mathbf{D}^2+\alpha_{1,s}(\mathbf{x},t_j)\mathbf{D}+\alpha_{0,s}(\mathbf{x},t_j)\right]\mathbf{U}_j^{(k)}-\sum_{q=0}^{M}d_{j,q}\mathbf{U}_q^{(k)}=\mathbf{R}_s^{(k)}(\mathbf{x},t_j). \tag{20}$$

By changing the indices, Eq. (20) can be written as

$$\left[\alpha_{2,s}(\mathbf{x},t_i)\mathbf{D}^2+\alpha_{2,s}(\mathbf{x},t_i)\mathbf{D}+\alpha_{0,s}(x,t_i)\right]\mathbf{U}_i^{(k)}-\sum_{j=0}^{M-1}d_{i,j}\mathbf{U}_j^{(k)}=\mathbf{R}_s^{(k)}(\mathbf{x},t_i)+d_{i,M}\mathbf{U}_{t_M}^{(k)}, \tag{21}$$

where

$$\mathbf{U}_{t_M}^{(1)}=f(x),\quad \text{for}\quad k=1\quad \text{and}\quad \mathbf{U}_{t_M}^{(k)}=\mathbf{U}_{t_0}^{(k-1)},\quad \text{for}\quad k=2,3,\ldots,P. \tag{22}$$

Eq. (21) constitutes an $M(N + 1) \times M(N + 1)$ matrix system given by

http://dx.doi.org/10.5772/64600

$$\begin{bmatrix} \mathbf{A}_{0,0} & \mathbf{A}_{0,1} & \mathbf{A}_{0,2} & \cdots & \mathbf{A}_{0,M-1} \\ \mathbf{A}_{1,0} & \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \cdots & \mathbf{A}_{1,M-1} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \mathbf{A}_{M-1,0} & \mathbf{A}_{M-1,1} & \mathbf{A}_{M-1,2} & \cdots & \mathbf{A}_{M-1,M-1} \end{bmatrix} \begin{bmatrix} \mathbf{U}_0^{(k)} \\ \mathbf{U}_1^{(k)} \\ \vdots \\ \mathbf{U}_{M-1}^{(k)} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_0^{(k)} \\ \mathbf{R}_1^{(k)} \\ \vdots \\ \mathbf{R}_{M-1}^{(k)} \end{bmatrix}, \tag{23}$$

where

$$\begin{aligned} &A_{i,i} = \alpha_{2,s}(\mathbf{x},t_i)\mathbf{D}^2 + \alpha_{2,s}(\mathbf{x},t_i)\mathbf{D} + \alpha_{0,s}(x,t_i) - d_{i,i}\mathbf{I}, \\ &A_{i,j} = -d_{i,j}\mathbf{I}, \quad i \neq j, \\ &\mathbf{R}_i^{(k)} = \mathbf{R}_s^{(k)}(\mathbf{x},t_i) + d_{i,M}\mathbf{U}_M^{(k)}, \\ &\alpha_{\mu,s}(\mathbf{x},t_i) = \begin{bmatrix} \alpha(\mathbf{x}_0,t_i) & & & \\ & \alpha(x_1,t_i) & & \\ & & \ddots & \\ & & & \alpha(x_N,t_i) \end{bmatrix}, \quad \mu = 0,1,2, \end{aligned} \tag{24}$$

and $\mathbf{I}$ is an identity matrix of size $(N+1) \times (N+1)$. The boundary conditions at the collocation points are

$$U^{(k)}(x_N,t_i) = g_a(t_i), \quad U^{(k)}(x_0,t_i) = g_b(t_i). \tag{25}$$

These boundary conditions are imposed on the main diagonal submatrices of the matrix system (23) to obtain a new system which takes the form

$$\begin{bmatrix} \hat{\mathbf{A}}_{0,0} & \hat{\mathbf{A}}_{0,1} & \hat{\mathbf{A}}_{0,2} & \cdots & \hat{\mathbf{A}}_{0,M-1} \\ \hat{\mathbf{A}}_{1,0} & \hat{\mathbf{A}}_{1,1} & \hat{\mathbf{A}}_{1,2} & \cdots & \hat{\mathbf{A}}_{1,M-1} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \hat{\mathbf{A}}_{M-1,0} & \hat{\mathbf{A}}_{M-1,1} & \hat{\mathbf{A}}_{M-1,2} & \cdots & \hat{\mathbf{A}}_{M-1,M-1} \end{bmatrix} \begin{bmatrix} \mathbf{U}_0^{(k)} \\ \mathbf{U}_1^{(k)} \\ \vdots \\ \mathbf{U}_{M-1}^{(k)} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{R}}_0^{(k)} \\ \hat{\mathbf{R}}_1^{(k)} \\ \vdots \\ \hat{\mathbf{R}}_{M-1}^{(k)} \end{bmatrix}, \tag{26}$$

where

$$\hat{A}_{i,i} = \left[\begin{array}{ccccc} 1 & 0 & 0 & \dots & 0 \\ \hline & & A_{i,i} & & \\ \hline 0 & 0 & 0 & \dots & 1 \end{array}\right], \quad \hat{A}_{i,j} = \left[\begin{array}{ccccc} 0 & 0 & 0 & \dots & 0 \\ \hline & & A_{i,i} & & \\ \hline 0 & 0 & 0 & \dots & 0 \end{array}\right], \quad i \neq j,$$
$$\hat{\mathbf{R}}_i = \left[\begin{array}{c} g_b(t_i) \\ \hline \mathbf{R}_i \\ \hline g_a(t_i) \end{array}\right]. \tag{27}$$

The matrix system (26) is solved for $U^{(k)}$, $k = 1, 2, \ldots, p$. The solutions at different subdomains are matched together along common boundaries to give the desired approximate solution. The patching condition is given by

$$u^{(k)}(x, t_{k-1}) = u^{(k-1)}(x, t_{k-1}), \tag{28}$$

which denotes the solution at the boundaries of the subintervals.

3. Numerical experimentation

In this section, we illustrate the practical applicability of the multidomain approach in solving nonlinear parabolic PDEs by considering the solutions of well-known nonlinear PDEs that have been reported in the literature.

Example 1. We consider the modified Burger's-Fisher equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{\partial^2 u}{\partial x^2} + u(1-u), \quad x \in (0,5), \quad t \in [0,10], \tag{29}$$

subject to boundary conditions

$$u(0,t) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{5t}{8}\right), \quad u(5,t) = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{5t}{8} - \frac{5}{4}\right), \tag{30}$$

and initial condition

$$u(x,0) = \frac{1}{2} - \frac{1}{2}\tanh\left(\frac{x}{4}\right). \tag{31}$$

The exact solution is given in [24] as

http://dx.doi.org/10.5772/64600

$$u(x,t)=\frac{1}{2}+\frac{1}{2}\tanh\left(\frac{5t}{8}-\frac{x}{4}\right). \tag{32}$$

Eq. (29) is an example of a generalized Burger's-Fisher equation that was solved in [4] using variational iteration method. Applying the QLM, we obtain the linearized system

$$\alpha_{2,s}(x,t)u''_{s+1}+\alpha_{1,s}(x,t)u'_{s+1}+\alpha_{0,s}(x,t)u_{s+1}-\dot{u}_{s+1}=R_s(x,t), \tag{33}$$

where

$$\alpha_{2,s}(x,t)=1,\quad \alpha_{1,s}(x,t)=-u_s,\quad \alpha_{0,s}(x,t)=-u'_s+1-2u_s,\quad R_s(x,t)=-u_su'_s-u_s^2. \tag{34}$$

In each subinterval $k = 1, 2, \ldots, p$, we must solve

$$\begin{aligned}&\alpha_{2,s}(x,t)\frac{\partial^2u^{(k)}_{s+1}}{\partial x^2}+\alpha_{1,s}(x,t)\frac{\partial u^{(k)}_{s+1}}{\partial x}+\alpha_{0,s}(x,t)u^{(k)}_{s+1}-\frac{\partial u^{(k)}_{s+1}}{\partial t}=R^{(k)}_s(x,t),\\&x\in(0,5),\quad t\in[t_{k-1},t_k],\end{aligned} \tag{35}$$

$$u^{(k)}(0,t)=\frac{1}{2}+\frac{1}{2}\tanh\left(\frac{5t}{8}\right),\quad u^{(k)}(5,t)=\frac{1}{2}+\frac{1}{2}\tanh\left(\frac{5t}{8}-\frac{5}{4}\right), \tag{36}$$

$$u^{(1)}(x,0)=\frac{1}{2}-\frac{1}{2}\tanh\left(\frac{x}{4}\right),\ k=0,\ \text{and}\ u^{(k)}(x,t_{k-1})=u^{(k-1)}(x,t_{k-1}),\ k=2,3,\ldots,p. \tag{37}$$

The matrices resulting from application of the spectral collocation in (33) are

$$\begin{aligned}&\mathbf{A}_{i,i}=\mathbf{D}^2+\alpha_{1,s}(\mathbf{x},t_i)\mathbf{D}+\alpha_{0,s}(\mathbf{x},t_i)-d_{i,i}I,\\&\mathbf{A}_{i,j}=-d_{i,j}I,\quad \text{when}\quad i\neq j,\\&\mathbf{B}^{(k)}_{i,s}=\mathbf{R}^{(k)}_{i,s}(\mathbf{x},t_i)+d_{i,M}\mathbf{U}^{(k)}_M,\end{aligned} \tag{38}$$

The initial condition at different subintervals is given by

$$\begin{aligned}&\mathbf{U}^{(1)}_{t_M}=f(x)=\frac{1}{2}-\frac{1}{2}\tanh\left(\frac{x}{4}\right),\quad \text{for}\quad k=1\quad \text{and}\\&\mathbf{U}^{(k)}_{t_M}=\mathbf{U}^{(k-1)}_{t_0},\quad \text{for}\quad k=2,3,\ldots,p.\end{aligned} \tag{39}$$

The boundary conditions at the collocation points are

$$U^{(k)}(x_N,t_i)=g_a(t_i)=\frac{1}{2}+\frac{1}{2}\tanh\left(\frac{5t_i}{8}\right),\ U^{(k)}(x_0,t_i)=g_b(t_i)=\frac{1}{2}+\frac{1}{2}\tanh\left(\frac{5t_i}{8}-\frac{5}{4}\right). \tag{40}$$

Making the relevant substitution, a matrix system similar to (26) is solved to obtain the approximate solution.

Example 2. We consider the modified Fitzhugh-Nagumo equation

$$\frac{\partial u}{\partial t}=\frac{\partial^2 u}{\partial x^2}+u(u-1)(1-u),\quad x\in(1,5),\quad t\in(0,1], \tag{41}$$

subject to boundary conditions

$$u(1,t)=\frac{1}{2}\left[1-\coth\left(\frac{-1}{2\sqrt{2}}+\frac{t}{4}\right)\right],\quad u(5,t)=\frac{1}{2}\left[1-\coth\left(\frac{-5}{2\sqrt{2}}+\frac{t}{4}\right)\right], \tag{42}$$

and initial condition

$$u(x,0)=\frac{1}{2}\left[1-\coth\left(\frac{-x}{2\sqrt{2}}\right)\right]. \tag{43}$$

The exact solution is given in [12]

$$u(x,t)=\frac{1}{2}\left[1-\coth\left(\frac{-x}{2\sqrt{2}}+\frac{t}{4}\right)\right]. \tag{44}$$

Eq. (41) is an example of a generalized Fitzhugh-Nagumo equation [12, 13]. Applying the QLM, we obtain a linearized system similar to that given in Eq. (33). The coefficients in this example are given by:

$$\alpha_{2,s}(x,t)=1,\quad \alpha_{1,s}(x,t)=0,\quad \alpha_{0,s}(x,t)=-1+4u_s-3u_s^2,\quad R_s(x,t)=2u_s^2-2u_s^3. \tag{45}$$

In each subinterval $k = 1, 2, \ldots, p$, we must solve:

http://dx.doi.org/10.5772/64600

$$\alpha_{2,s}(x,t)\frac{\partial^2 u_{s+1}^{(k)}}{\partial x^2}+\alpha_{1,s}(x,t)\frac{\partial u_{s+1}^{(k)}}{\partial x}+\alpha_{0,s}(x,t)u_{s+1}^{(k)}-\frac{\partial u_{s+1}^{(k)}}{\partial t}=R_s^{(k)}(x,t),\quad x\in(1,5),\quad t\in(t_{k-1},t_k], \tag{46}$$

$$u^{(k)}(1,t)=\frac{1}{2}\left[1-\coth\left(\frac{-1}{2\sqrt{2}}+\frac{t}{4}\right)\right],\quad u^{(k)}(5,t)=\frac{1}{2}\left[1-\coth\left(\frac{-5}{2\sqrt{2}}+\frac{t}{4}\right)\right], \tag{47}$$

$$u^{(1)}(x,0)=\frac{1}{2}\left[1-\coth\left(\frac{-x}{2\sqrt{2}}\right)\right],\ k=0,\ \text{and}\ u^{(k)}(x,t_{k-1})=u^{(k-1)}(x,t_{k-1}),\quad k=2,3,\ldots,p. \tag{48}$$

The application of the spectral collocation in (33) results into the following set of coefficient matrices:

$$\begin{aligned}\mathbf{A}_{i,i}&=\mathbf{D}^2+\alpha_{0,s}(\mathbf{x},t_i)-d_{i,i}I,\\ \mathbf{A}_{i,j}&=-d_{i,j}I,\quad \text{when}\quad i\neq j,\\ \mathbf{B}_{i,s}^{(k)}&=\mathbf{R}_{i,s}^{(k)}(x,t_i)+d_{i,M}\mathbf{U}_M^{(k)},\end{aligned} \tag{49}$$

The initial condition at different subintervals is given by:

$$\mathbf{U}_{t_M}^{(1)}=f(x)=\frac{1}{2}-\frac{1}{2}\tanh\left(\frac{x}{4}\right),\ \text{for}\ k=1\ \text{and}\ \mathbf{U}_{t_M}^{(k)}=\mathbf{U}_{t_0}^{(k-1)},\ \text{for}\ k=2,3,\ldots,p. \tag{50}$$

The boundary conditions at the collocation points are given by:

$$\begin{aligned}U^{(k)}(x_N,t_i)&=g_a(t_i)=\frac{1}{2}\left[1-\coth\left(\frac{-1}{2\sqrt{2}}+\frac{t_i}{4}\right)\right],\\ U^{(k)}(x_0,t_i)&=g_b(t_i)=\frac{1}{2}\left[1-\coth\left(\frac{-5}{2\sqrt{2}}+\frac{t_i}{4}\right)\right].\end{aligned} \tag{51}$$

Example 3. We consider the modified Burger's-Huxley equation:

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}=\frac{\partial^2 u}{\partial x^2}+u(1-u)(u-0.1),\quad x\in(0,1),\quad t\in[0,10], \tag{52}$$

subject to boundary conditions

$$u(0,t)=\frac{1}{2}-\frac{1}{2}\tanh\left[\frac{1}{2}\left(-0.9t\right)\right],\quad u(1,t)=\frac{1}{2}-\frac{1}{2}\tanh\left[\frac{1}{2}\left(1-0.9t\right)\right], \tag{53}$$

and initial condition

$$u(x,0)=\frac{1}{2}-\frac{1}{2}\tanh\left(\frac{x}{2}\right). \tag{54}$$

The exact solution is given in [25] as

$$u(x,t)=\frac{1}{2}-\frac{1}{2}\tanh\left[\frac{1}{2}\left(x-0.9t\right)\right]. \tag{55}$$

Eq. (52) is an example of a generalized Burger's-Huxley equation [25]. Applying the QLM, we obtain a linearized system similar to that given in Eq. (33). The coefficients in this example are given by

$$\begin{aligned}&\alpha_{2,s}(x,t)=1,\quad \alpha_{1,s}(x,t)=-u_s,\quad \alpha_{0,s}(x,t)=-0.1-u_s'+2.2u_s-3u_s^2,\\&R_s(x,t)=-u_su_s'+1.1u_s^2-2u_s^3.\end{aligned} \tag{56}$$

In each subinterval $k = 1, 2, \ldots, p$, we must solve

$$\begin{aligned}&\alpha_{2,s}(x,t)\frac{\partial^2u_{s+1}^{(k)}}{\partial x^2}+\alpha_{1,s}(x,t)\frac{\partial u_{s+1}^{(k)}}{\partial x}+\alpha_{0,s}(x,t)u_{s+1}^{(k)}-\frac{\partial u_{s+1}^{(k)}}{\partial t}=R_s^{(k)}(x,t),\\&x\in(0,1),\quad t\in(t_{k-1},t_k],\end{aligned} \tag{57}$$

$$u^{(k)}(0,t)=\frac{1}{2}-\frac{1}{2}\tanh\left[\frac{1}{2}\left(-0.9t\right)\right],\quad u^{(k)}(1,t)=\frac{1}{2}-\frac{1}{2}\tanh\left[\frac{1}{2}\left(1-0.9t\right)\right], \tag{58}$$

$$u^{(1)}(x,0)=\frac{1}{2}-\frac{1}{2}\tanh\left(\frac{x}{2}\right),\ k=0,\ \text{and}\ \ u^{(k)}(x,t_{k-1})=u^{(k-1)}(x,t_{k-1}),\ k=2,3,\ldots,p. \tag{59}$$

The application of the spectral collocation in (33) results into the following set of coefficient matrices

http://dx.doi.org/10.5772/64600

$$\begin{aligned}
&\mathbf{A}_{i,i} = \mathbf{D}^2 + \alpha_{1,s}(\mathbf{x},t_i)\mathbf{D} + \alpha_{0,s}(\mathbf{x},t_i) - d_{i,i}I,\\
&\mathbf{A}_{i,j} = -d_{i,j}I, \quad \text{when} \quad i \neq j,\\
&\mathbf{B}_{i,s}^{(k)} = \mathbf{R}_{i,s}^{(k)}(\mathbf{x},t_i) + d_{i,M}\mathbf{U}_M^{(k)},
\end{aligned} \tag{60}$$

The initial condition at different subintervals is given by

$$\mathbf{U}_{t_M}^{(1)} = f(x) = \frac{1}{2} - \frac{1}{2}\tanh\left(\frac{x}{2}\right), \text{ for } k = 1 \text{ and } \mathbf{U}_{t_M}^{(k)} = \mathbf{U}_{t_0}^{(k-1)}, \text{ for } k = 2,3,\ldots,p. \tag{61}$$

The boundary conditions at the collocation points are given by

$$\begin{aligned}
U^{(k)}(x_N,t_i) &= g_a(t_i) = \frac{1}{2} - \frac{1}{2}\tanh\left[\frac{1}{2}\left(-0.9t_i\right)\right],\\
U^{(k)}(x_0,t_i) &= g_b(t_i) = \frac{1}{2} - \frac{1}{2}\tanh\left[\frac{1}{2}\left(1-0.9t_i\right)\right].
\end{aligned} \tag{62}$$

4. Results and discussion

In this section, we present the results for the absolute error values at selected values of x and t and the computational time that is obtained when Examples 1–3 are solved using both the bivariate spectral collocation method (single-domain approach) and the multidomain bivariate spectral collocation method. The absolute error is evaluated as

$$Abs_{x_i,t_j} = \left|u_e(x_i,t_j) - u_a(x_i,t_j)\right|, \ 0 \le i \le N, 0 \le j \le M, \tag{63}$$

where $u_e(x_i, t_j)$ is the exact solution and $u_a(x_i, t_j)$ is the approximate solution at the collocation points (x_i, t_j). In each example, two tables have been presented to compare the performance of the two approaches that are used in solving each example. The results indicate that multidomain bivariate spectral collocation method is very accurate and the results are generated faster when compared to solving the same problem over single domain. The results obtained from approximating the solution of (29) are given below. **Table 1** shows the results generated when the bivariate spectral collocation method (single domain) is used whereas **Table 2** presents the results obtained when using the multidomain bivariate spectral collocation method. The bivariate spectral collocation method gives on average absolute errors of 10^{-6} whereas those obtained when the multidomain bivariate spectral collocation method is used are 10^{-12}, an indication that the multidomain approach is more accurate. The computational time in the case of the multidomain approach is lesser (0.019602 sec) than (0.103606 sec) that is obtained when solving the problem over single domain.

x		t			
		2.0	**4.0**	**6.0**	**8.0**
0.4775		4.91994e-007	3.31285e-007	1.99894e-008	6.67958e-008
1.3650		4.36608e-007	7.17567e-007	1.76673e-008	1.70637e-007
2.5000		3.26724e-006	1.43695e-006	1.42849e-007	2.19591e-007
3.6350		2.35521e-007	2.54426e-006	1.48557e-007	6.72760e-007
4.5225		7.14695e-006	6.81188e-006	2.08164e-006	2.15409e-007
CPU time (sec)	0.103606				

Table 1. MatLab solution: Absolute error values obtained when solving Example 1 using $N = M = 20$, single domain, with Lagrange basis and Gauss-Lobatto nodes, iterations=10.

		t			
x		**2.0**	**4.0**	**6.0**	**8.0**
0.4775		4.77396e-014	7.99361e-015	1.16573e-014	2.66454e-014
1.3650		7.41074e-013	1.29896e-014	1.13243e-014	2.81997e-014
2.5000		3.31513e-013	3.83027e-014	1.25455e-014	2.44249e-015
3.6350		3.37175e-012	7.40519e-014	8.88178e-015	4.21885e-015
4.5225		2.55729e-012	1.52323e-013	2.68674e-014	2.10942e-014
CPU time (sec)	0.019602				

Table 2. MatLab solution: Absolute error values obtained when Example 1 is solved using $N = 20$, $M = 5$, $p = 10$, with Lagrange basis and Gauss-Lobatto nodes, iterations=10.

The results obtained from approximating the solution of (41) are given in **Tables 3** and **4**. **Table 3** shows the results generated when the bivariate spectral collocation method (single domain) is used whereas **Table 4** presents the results obtained when using the multidomain bivariate spectral collocation method. The results are similar to those of Example 1, thus the multidomain approach is more efficient than single-domain approach when it is applied in solving nonlinear parabolic PDEs defined over large-time domain.

		t			
x		**0.2**	**0.4**	**0.6**	**0.8**
1.0039		1.35033e-006	1.14349e-005	1.42724e-005	2.50691e-006
1.9283		8.19001e-008	1.97335e-008	4.17650e-008	4.41645e-009
3.0000		6.03945e-009	1.22066e-008	9.50467e-009	8.55427e-009
4.0717		8.95142e-011	1.11732e-009	2.53453e-009	3.16956e-009
4.9372		2.23332e-012	2.90505e-011	1.05300e-010	1.94311e-010
CPU time (sec)	0.135906				

Table 3. MatLab solution: Absolute error values obtained when solving Example 2 using $N = 50$, $M = 10$, single domain, with Lagrange basis and Gauss-Lobatto nodes, iterations=10.

http://dx.doi.org/10.5772/64600

			t		
x		**0.2**	**0.4**	**0.6**	**0.8**
1.0039		7.57705e-012	3.20854e-012	6.38556e-012	2.23954e-011
1.9283		8.88178e-015	2.10942e-014	3.37508e-014	2.22045e-015
3.0000		1.55431e-015	7.32747e-015	1.77636e-014	1.95399e-014
4.0717		6.83897e-014	5.66214e-014	8.79297e-014	3.50830e-014
4.9372		1.11511e-012	2.31637e-012	7.83817e-014	4.10783e-014
CPU time (sec)	0.026619				

Table 4. MatLab solution: Absolute error values obtained when Example 2 is solved using $N = 50$, $M = 5$, $p = 100$, with Lagrange basis and Gauss-Lobatto nodes, iterations=10.

The results obtained from approximating the solution of (52) are given below. **Table 5** shows the results generated when the bivariate spectral collocation method (single domain) is used whereas **Table 6** presents the results obtained when using the multidomain bivariate spectral collocation method. The results indicate that the multidomain approach is very accurate and computationally faster when it is applied to solve nonlinear PDEs defined over large-time intervals.

			t		
x		**2.0**	**4.0**	**6.0**	**8.0**
0.0010		1.61398e-005	9.61285e-005	7.30576e-005	7.70011e-006
0.2321		2.99891e-005	1.09535e-004	9.97674e-005	2.21837e-006
0.5000		2.15105e-00	9.64207e-005	1.07763e-004	7.47171e-006
0.7679		1.07315e-005	1.90612e-005	6.33297e-005	5.44788e-006
0.9843		3.75286e-005	1.10078e-004	2.79722e-005	6.65481e-007
CPU time (sec)	0.026619				

Table 5. MatLab solution: Absolute error values obtained when solving Example 3 using $N = 50$, $M = 10$, single domain, with Lagrange basis and Gauss-Lobatto nodes, iterations=10.

			t		
x		**2.0**	**4.0**	**6.0**	**8.0**
0.0010		5.21284e-010	4.77699e-010	3.01078e-010	3.88833e-010
0.2321		2.05589e-011	4.15302e-011	4.59769e-011	2.75941e-011
0.5000		2.27406e-011	1.40028e-011	1.33067e-011	1.88674e-011
0.7679		5.37154e-011	3.15495e-011	1.85438e-011	2.47591e-012
0.9843		2.02505e-010	6.77300e-011	2.10308e-010	1.70352e-010
CPU time (sec)	0.028104				

Table 6. MatLab solution: Absolute error values obtained when Example 3 is solved using $N = 50$, $M = 5$, $p = 100$, with Lagrange basis and Gauss-Lobatto nodes, iterations=10.

The lesser computational time that is evident in the case when the multidomain approach is applied to solve the nonlinear PDE is attributed to the fact that the multidomain approach uses very few number of collocation points in each subinterval for the time variable than in the single-domain approach. This reduction in the number of collocation points significantly reduces the size of the resulting coefficient matrices. The small -sized coefficient matrices are

less dense and take less CPU time to produce results. The high accuracy and less computational time substantiate our claim that the multidomain bivariate spectral collocation method is a powerful numerical method for solving nonlinear parabolic PDEs that are defined over large-time intervals. The QLM is a powerful technique for simplifying nonlinear PDEs as very accurate results are obtained after 10 iterations only. The spectral collocation-based methods yield very accurate results with a few number of grid points as the approximate solution that is searched for is a higher degree polynomial. In the numerical experimentation, the symmetrically distributed Gauss-Lobatto (G-L) collocation points have been used instead of equispaced grid points as the G-L nodes have a feature that tends to uniformly distribute the approximation errors across the entire interval of approximation [26]. The equispaced nodes, on the other hand, produce oscillations near the end of interval of approximation, a behavior referred to as Runge phenomena [27].

5. Conclusion

The multidomain bivariate spectral collocation method has been used successfully to solve nonlinear parabolic PDEs that arise in a wide range of applications like genetics, biology, heat and mass transfer and wave processes. The approximate results confirm that the multidomain bivariate spectral collocation method is very accurate and computationally faster when it is used to solve nonlinear parabolic PDEs that are defined over large-time domains. This approach is an alternative to other numerical methods that can be used to solve nonlinear parabolic partial differential equations. The multidomain bivariate spectral collocation method being more accurate and computationally faster can therefore be adopted and extended to solve similar problems that model real-life phenomenon.

Acknowledgements

This work is based on the research supported in part by the National Research Foundation of South Africa (Grant No. 85596).

Author details

Motsa Sandile Sydney[1*], Samuel Felix Mutua[1] and Shateyi Stanford[2]

*Address all correspondence to: sandilemotsa@gmail.com

1 School of Mathematics, Statistics & Computer Science, University of KwaZulu-Natal, Scottsville, Pietermaritzburg, South Africa

2 University of Venda, Thohoyandou, Limpopo Province, South Africa

http://dx.doi.org/10.5772/64600

References

[1] A. H. Khater, R. S. Temsah, Numerical solutions of some nonlinear evolution equations by Chebyshev spectral collocation methods, Int J Comp Math, Vol. 84, pp. 326–339, 2007.

[2] X. Y. Wang, Exact and explicit solitary wave solutions for the generalised Fishers equation, Phys Lett A, Vol. 131, pp. 277–291, 1988.

[3] W. Hereman, M. Takaoka, Solitary wave solutions of nonlinear evolution and wave equations using a direct method and MACSYMA, J Phys A, Vol. 23, pp. 34–48, 1990.

[4] M. A. Abdou, A. A. Soliman, Variational iteration method for solving Burger's and coupled Burger's equation, J Comput Appl Math, Vol. 181, pp. 51–62, 2005.

[5] R. Jiwari, Quasilinearization approach for numerical simulation of Burger's equation, Comp Phys Commun, Vol. 183, pp. 2413–2423, 2012.

[6] R. Jiwari, Mittal R. C., Sharma K. K., A numerical based on weighed average differential quadrature method for the numerical solution of Burger's equation, Appl Math Comput, Vol. 219, pp. 6680–6691, 2013.

[7] M. Javidi, A numerical solution of the generalized Burger's-Huxley equation by pseudospectral method and Darvishi's preconditioning, Appl Math Comput, Vol. 175, pp. 16–28, 1990.

[8] M. Javidi, A numerical solution of the generalized Burger's-Huxley equation by pseudospectral method, Appl Math Comput, Vol. 15, pp. 99–108, 1990.

[9] I. Hashim, M. S. Noorani, M. R. Al-Hadidi, Solving the generalized Burger's-Huxley equation using the Adomian decomposition method, Math Comput Model, Vol. 16, pp. 11–19, 2006.

[10] R. C. Mittal, R. Jiwari, Study of Burger-Huxley equation by differential quadrature method, Int J Appl Math Mech, Vol. 5, pp. 1–9, 2009.

[11] A. J. Khattak, A computational meshless method for the generalised Burger's-Huxley equation, Appl Math Model, Vol. 33, pp. 3718–3729,2006

[12] T. Kawahara, M. Tanaka, Interactions of traveling fronts: an exact solution of a nonlinear diffusion equations, Phys Lett, Vol. 97, pp. 311, 1983.

[13] M. C. Nucci, P. A. Clarkson, The nonclassical method is more general than the direct method for symmetry reductions. An example of the Fitzhugh-Nagumo equation, Phys Lett A, Vol. 164, pp. 49–56, 1992.

[14] J. H. He, Homotopy perturbation method for bifurcation of nonlinear problems, Int J Nonlinear Sci Numer Simul, Vol. 6, pp. 27–33, 2005.

[15] L. N. Trefethen, Spectral Methods in MATLAB, SIAM, 2000.

[16] S. S. Motsa, V. M. Magagula, P. Sibanda, A Bivariate Chebyshev Spectral Collocation Quasilinearization Method for Nonlinear Evolution Parabolic Equations, Sci World J, Vol. 2014, pp 13, 2014. doi:10.1155/2014/581987

[17] S. Ismail, On bivariate polynomial interpolation, East J Approx Vol. 8, pp. 209–218, 2002.

[18] R. E. Bellman, R. E. Kalaba, Quasilinearization and Nonlinear Boundary-Value Problems, Elsevier Publishing Company, New York, 1965.

[19] S. S. Motsa, P. Dlamini, M. Khumalo, A new multi-stage spectral relaxation method for solving chaotic initial value systems, Nonlinear Dynam, Vol. 72, Issue 1–2, pp. 265–283, 2013.

[20] S. S. Motsa, A new piecewise-quasilinearization method for solving chaotic systems of initial value problems, Cent Eur J Phys, Vol. 10, Issue 4, pp. 936–946, 2012.

[21] S. S. Motsa, A new piece-wise-quasilinearisation method approach to a four-dimensional hyper-chaotic system with cubic nonlinearity, Nonlinear Dynam Vol. 70, pp. 651–657, 2012.

[22] V. Lakshmikantham, An extension of the method of quasilinearization, J Optim Theory Appl Vol. 82, pp. 315–321, 1994.

[23] V. Lakshmikantham, Further improvement of generalized quasilinearization, Nonlinear Anal Vol. 27, pp. 315–321, 1996.

[24] W. Wang, A. J. Roberts, Diffusion approximation for self-similarity of stochastic advection in Burger's equation, Commun Math Phys, Vol. 5, pp. 37–48, 2014.

[25] O. Y. Yetimova, N. A. Kudryashov, Exact solutions of the Burgers-Huxley equation, J Appl Math Mech, Vol. 68, Issue 3, pp. 413–420, 2004.

[26] Q. Chen, Ivo BabuSkab, Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle, Comput Methods Appl Mech Eng Vol. 128, pp. 405–417, 1995.

[27] J. F. Epperson, On the Runge example, Amer Math Monthly, Vol. 94, pp. 329–341, 1987.

Chapter 2

Feedback and Partial Feedback Linearization of Nonlinear Systems: A Tribute to the Elders

Issa Amadou Tall

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64689

Abstract

Arthur Krener and Roger Brockett pioneered the feedback linearization problem for control systems, that is, the transforming of a nonlinear control system into linear dynamics via change of coordinates and feedback. While the former gave necessary and sufficient conditions to linearize a system under change of coordinates only, the latter introduced the concept of feedback and solved the problem for a particular case. Their work was soon extended in the earlier eighties by Jakubczyk and Responder, and Hunt and Su who gave the conditions for a control system to be linearizable by change of coordinates and feedback (full rank and involutivity of the associated distributions). It turned out that those conditions are very restrictive; however, it was showed later that systems that fail to be linearizable can still be transformed into two interconnected subsystems: one linear and the other nonlinear. This fact is known as partial feedback linearization. For input-output systems with well-defined relative degree, coordinates can be found by differentiating the outputs. For systems without outputs, necessary and sufficient geometric conditions for partial linearization have been obtained in terms of the Lie algebra of the system; however, both results of linearization and partial feedback linearization lack practicability. Until recently, none has provided a way to actually compute the linearizing coordinates and feedback. In this paper, we propose an algorithm allowing to find the linearizing coordinates and feedback if the system is linearizable, and in the contrary, to decompose a system (without outputs) while achieving the largest linear subsystem. Those algorithms are built upon successive applications of the Frobenius theorem. Examples are provided to illustrate.

Keywords: feedback, Frobenius theorem, partial linearization

1. Introduction

Roger Brockett is considered as the father of feedback linearization, one of the most important techniques for studying nonlinear systems. The problem of feedback linearization seeks to find new coordinates in which the system exhibits linear dynamics driven by new control inputs. The role of linear systems in engineering and mechanical systems has already been demonstrated in several applications. First, let us consider a linear system

$$\Lambda : \begin{cases} \dot{x} = Fx + Gu = Fx + G_1u_1 + \cdots + G_mu_m \\ y = Hx = H_1x_1 + \cdots + H_nx_n \end{cases} \tag{1}$$

where $Fx, G_1, \cdots, G_m$ are, respectively, on $\Re^n$, Hx a linear vector field on $\Re^p$, $x = (x_1, \cdots, x_n) \epsilon \Re^n$ denotes the state of the system, and $u = (u_1, \cdots, u_m) \epsilon \Re^m$ the control input. To the linear system Λ, we attach two geometric objects: one called controllability space $\mathcal{C}_n = span\,[G\ FG\ \cdots F^{n-1}G]$ as a $n \times (nm)$ matrix whose columns are those of the matrices $F^{i-1}G$ for $i = 1,\ \ 2, \cdots, n$; the other called observability space $\mathcal{O}_n = span\,[H\ HF\ \cdots H^{n-1}F]$ as a $p \times (nm)$ matrix whose columns are those of the matrices $H^{i-1}F$ for $i = 1,\ \ 2, \cdots, n$. The system Λ is controllable if and only if $\dim \mathcal{C}_n = n$ and the system is observable if and only if $\dim \mathcal{O}_n = p$. By a linear change of coordinates $z = Tx$ and a linear feedback $u = Kx + Lv$ where T, K, and L are matrices of appropriate sizes, T and L being invertible, the system Λ is transformed into a linear equivalent one

$$\overline{\Lambda} : \begin{cases} \dot{z} = Az + Bv = Az + B_1v_1 + \cdots + B_mv_m \\ y = Cz = C_1z_1 + \cdots + C_nz_n \end{cases} \tag{2}$$

with $A = \mathrm{T(F + GK)}T^{-1}, B = TGL$, and $C = HT^{-1}$.

For the linear system $\dot{x} = Ax + Bu$ where A and B are $n \times n$ and $n \times m$ matrices, respectively, we denote by $M^j = [A\ \ AB\ \ \cdots A^{j-1}B]$ and $m_j = \dim M^j$. We define $k_j = \max\{n_i \mid n_i \geq j\}$ where $n_0 = 0$ and $n_i = m_i - m_{i-1}$ for $1 \leq i \leq n$. It is straightforward to notice that $k_1 \geq \cdots \geq k_m$ with $k_1 + \cdots + k_m = n$. It is a classical result of the linear control theory that a certain choice of the matrices T, K, and L leads to the Brunovsky canonical form $\overline{\Lambda} = \Lambda_{BR}$ for which $A = diag\ \{A_1,\ \ A_2, \cdots, A_m\}$ and $B = diag\ \{b_1,\ \ b_2, \cdots, b_m\}$ with (see [1])

http://dx.doi.org/10.5772/64689

$$A\_i = \begin{pmatrix} 0 & 1 & 0\cdots & 0 \\ 0 & 0 & 1\cdots & 0 \\ \vdots & \vdots & \vdots\ddots & \vdots \\ 0 & 0 & 0\cdots & 1 \\ 0 & 0 & 0\cdots & 0 \end{pmatrix} \text{ and } b\_i = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{3}$$

form a canonical pair of dimension k_i. Moreover, $C = (1\ 0\ \cdots\ 0)$.

Now let us consider a nonlinear control system (control-affine for simplicity)

$$\acute{O}: \begin{cases} \dot{x} = f(x) + g(x)u = f(x) + g_1(x)u_1 + \cdots + g_m(x)u_m, x\grave{o}\Re^n, u\grave{o}\Re^m \\ y = h(x) = (h_1(x), \cdots, h_p(x)) \end{cases} \tag{4}$$

where $x = (x_1, \cdots, x_n)\ \epsilon\ \Re^n$ denotes the state of the system, and $u = (u_1, \cdots, u_m)\ \epsilon\ \Re^m$ the control input., $f, g_1, \cdots, g_m$ are smooth or analytic vector fields with $f(0) = 0$, and $h_1, \cdots, h_p$ analytic functions on $\Re^n$.

The problem of finding new coordinates $z = (z_1, \cdots, z_n)\ \epsilon\ \Re^n$ in which the system Σ, driven by new inputs $v = (v_1, \cdots, v_m)\ \epsilon\ \Re^m$, takes the form $\overline{\Lambda}$ is referred as the input-output static state feedback linearization. For input-output systems, the problem of linearization is equivalent to achieving a relative degree (see details later). When the relative degree is achieved, finding the coordinates system in which the system becomes linear is a simple differentiation process. For systems without outputs, we only refer to static state linearization (Problem 1) or static state feedback linearization (Problem 2) as follows:

Problem 1: Find new coordinates $z = \Phi(x)$ that transform the system $\Sigma\ :\ \dot{x} = f(x) + g(x)u$ into a linear controllable system $\dot{z} = Az + Bu$.

Problem 2: Find new coordinates $z = \Phi(x)$ and an invertible feedback $u = \alpha(x) + \beta(x)v$ that transform the system $\Sigma\ :\ \dot{x} = f(x) + g(x)u$ into a linear controllable system $\dot{z} = Az + Bv$.

Arthur Krener [2] formulated and completely solved the first problem by showing that the Lie brackets of some vector fields have to be zero, that is, a certain set of vector fields associated with the system have to commute. Roger Brockett [3] solved the second problem under the assumption that $m = 1$ (single-input), $p = 1$ (single-output) and β is constant. The general case of input-output feedback linearization (Problem 2) was solved by Jakubczyk and Respondek [4] on one side, and independently by Hunt and Su [5] on the other side. Necessary and sufficient geometric conditions were obtained and showed that there is only a small class of nonlinear systems that can be linearized by feedback. Indeed, the system should satisfy the following two strong conditions:

(F1) an involutive distribution,

(F2) a distribution with full rank equal to the dimension of the system.

Those conditions are very restrictive, thus making the class of nonlinear systems that can be linearized by static state feedback very small. To enlarge the class of nonlinear systems that can be analyzed via feedback linearization, several techniques have been introduced including dynamic feedback linearization, nonregular state feedback linearization, partial feedback linearization, orbital feedback linearization, and transverse feedback linearization. Dynamic feedback linearization differs from static state feedback linearization in the sense that a compensator $\dot{w} = a(x,w) + b(x,w)v, w\epsilon\Re^q, u = \alpha(x,w) + \beta(x,w)v$ is thought that enlarges the dimension of the system. This means that one tries to linearize the system

$$\Sigma : \begin{cases} \dot{x} = f(x) + g(x)\alpha(x,w) + g(x)\beta(x,w)v, \ x\epsilon\,\Re^n, v\,\epsilon\,\Re^m \\ \dot{w} = a(x,w) + b(x,w)v, \quad w\epsilon\,\Re^q \end{cases} \tag{5}$$

using an extended state space transformation $z = \varphi(x,w)\epsilon\Re^{n+q}$. This problem is referred as regular feedback linearization ($\beta(.)$ is an invertible matrix). More general feedbacks have been exploited to enlarge the class of linearizable systems by allowing the matrix $\beta(.)$ to be noninvertible, that is, admitting fewer inputs than the original system [6, 7]. In this case, we talk about nonregular feedback linearization [8]. Orbital feedback linearization, also known as time scale feedback linearization, introduces a new time scale τ such that $dt/d\tau = \gamma(x)$ is a positive function (preserve orientation). Hence, in the new time scale τ, the problem becomes to linearize the time-scaled system (see [9] and references therein)

$$\Sigma : \begin{cases} \frac{dx}{d\tau} = \gamma(x)f(x) + g(x)u = \gamma(x)f(x) + g_1(x)u_1 + \cdots + g_m(x)u_m, \ x\epsilon\,\Re^n, u\,\epsilon\,\Re^m \\ y = h(x) = (h_1(x), \cdots, h_p(x)) \end{cases} \tag{6}$$

Transverse feedback linearization [10] deals with transforming a control-affine system coupled with a controlled invariant manifold into a system whose dynamics, transversal to the invariant manifold, are linear and controllable.

The feedback linearization problem has been thoroughly investigated in the past four decades but have regained interest recently with new algorithms developed to circumvent the solving of partial differential equations associated to the linearization (see [4, 5, 21–28], and the references therein). Whenever a system fails to satisfy either condition (F1) or (F2), its dynamics contain nonlinearities in any given coordinate system. The fundamental question is in which coordinates does the system exhibit the largest linear subsystem. This question was first addressed naturally for systems with outputs [6, 7, 11–20]. We propose in this paper an algorithmic way of transforming a control system into a cascade of two systems: one nonlinear and one linear with the largest dimension. First, we will recall basics about vector fields and the Frobenius theorem, then Section 3 deals with linearization of control systems with outputs,

Section 4 contains the partial linearization algorithm. We end the paper with Section 5 with few examples as an illustration.

2. Vector fields and Frobenius theorem

The theory of differential equations is one of the most productive and useful contributions of our modern times. Its applications are widespread in all branches of natural sciences, particularly, in physics, biology, chemistry, engineering, ecology, and in weather predictions, just to name few. It plays the role of a connector between abstract mathematical theories and applications in real world problems. Paraphrasing Newton quoted as saying that "it is useful to solve differential equations," a lot has been deserved in solving differential equations with various methods and techniques provided in the literature. Existence and uniqueness of solutions have been addressed in many scientific papers and textbooks. Consider the simplest expression of a linear partial differential equation

$$f_1(x)\frac{\partial u}{\partial x_1} + \cdots + f_n(x)\frac{\partial u}{\partial x_n} = b(x) \tag{7}$$

where $f_1(x), \cdots, f_n(x)$, and $b(x)$ are smooth or analytic functions in the variable x. This partial differential equation is referred to as a homogeneous (resp. nonhomogeneous) linear first order partial differential equation when $b(x) \equiv 0$ (resp. $b(x) \not\equiv 0$). The vector field $f(x)$ whose components are $f_1(x), \cdots, f_n(x)$ is called the characteristic vector field of the homogeneous equation and the corresponding dynamical system $\dot{x} = f(x)$, its characteristic equation. The solutions of the system are the integral curves of the characteristic equation and are often obtained by solving the so-called Lagrange subsidiary equation (also called characteristic equation)

$$\frac{dx_1}{f_1(x)} = \cdots = \frac{dx_n}{f_n(x)} = \frac{du}{b(x)} \tag{8}$$

Several methods have been devoted to the solving of such system among them Euler's method and Natani's method. Most of the work on ordinary differential equations have been done around equilibrium points (nonregular or singular point), that is, a point x_0 where ($fx_0 = 0$) . The reason being that regular points, that is, where $f(x_0) \neq 0$ are not topologically reach, because in those neighborhoods all trajectories are straight parallel lines (straightening theorem). Though this fact remains true and hence often neglected, the straightening theorem has many important applications. Indeed, a solution of the nonhomogeneous partial differential equation above can be easily found around a regular point x_0 of f by simple quadrature in new coordinates: If $z = \varphi(x)$ is a change of coordinates around x_0 that rectifies the vector

field f, that is, such that $\varphi_*(f) = \frac{\partial}{\partial z_n}$, then the nonhomogeneous equation simplifies as $\frac{\partial \tilde{u}}{\partial z_n} = \tilde{b}(z)$, where $u(x) = \tilde{u}(\varphi(x))$ and $b(x) = \tilde{b}(\varphi(x))$. A solution $\tilde{u}$ (yielding $u = \tilde{u} \circ \varphi$) is given

$$\tilde{u}(z) = a(z_1, \cdots, z_{n-1}) + \int_0^{z_n} \tilde{b}(z_1, \cdots, z_{n-1}, \varepsilon) d\varepsilon. \tag{9}$$

The dynamical system $\dot{x} = f(x)$ takes in this case the canonical form

$$\begin{cases} \dot{z}_1 = 0 \\ \dot{z}_2 = 0 \\ \quad \vdots \\ \dot{z}_{n-1} = 0 \\ \dot{z}_n = 1 \end{cases} \tag{10}$$

Theorem 1: (Flow-box) Let f be a vector field defined in a neighbourhood of a nonsingular point $x_0 \in \Re^n$, that is, $f(x_0) \neq 0$. There exists a local change of coordinates $z = \Phi(x)$ in a neighbourhood $\mathcal{U}$ of x_0 such that $\Phi_*(f)(z) = \frac{\partial}{\partial x_n}$ for all $z \in \mathcal{U}$.

The existence and proof of this theorem, as well as its general form, can be found in the literature. The only difficulty in applying the straightening theorem is in finding the straightening diffeomorphism as one needs to solve the system of highly nonlinear partial differential equations:

$$\begin{cases} \frac{\partial \Phi_1}{\partial x_1} f_n(x) + \cdots + \frac{\partial \Phi_1}{\partial x_n} f_n(x) = 0 \\ \frac{\partial \Phi_2}{\partial x_1} f_n(x) + \cdots + \frac{\partial \Phi_2}{\partial x_n} f_n(x) = 0 \\ \qquad \vdots \\ \frac{\partial \Phi_{n-1}}{\partial x_1} f_n(x) + \cdots + \frac{\partial \Phi_{n-1}}{\partial x_n} f_n(x) = 0 \\ \frac{\partial \Phi_n}{\partial x_1} f_n(x) + \cdots + \frac{\partial \Phi_n}{\partial x_n} f_n(x) = 1 \end{cases} \tag{11}$$

In earlier work [25], we provided a solution to this problem by giving explicit changes of coordinates, which will be recalled below. If x_0 is a singular point, that is, $f(x_0) = 0$, the notion of linearization, and later of normal form, were introduced by Poincare. Before we recall those facts, let us remind the reader that dynamical systems are a subclass of a largest class named control systems. Indeed, a control system can be interpreted as a parameterized family of dynamical systems $\dot{x} = F(x, u)$ where for each fixed value of u, $F_u : x \to F_u(x) = F(x, u)$ is a vector field. When $u = 0$, we rediscover dynamical systems. Poincare was the first to address the

http://dx.doi.org/10.5772/64689

problem of linearization for dynamical systems around an equilibrium point. He indeed showed that when $\frac{\partial f}{\partial x}(x_0) = F$ is a matrix whose spectrum $\lambda = (\lambda_1, \cdots, \lambda_n)$ is not resonant, then new coordinates $z = \varphi(x)$ exist where the dynamical system takes the linear form $\dot{z} = Fz$. We recall that a spectrum $\lambda = (\lambda_1, \cdots, \lambda_n)$ is called resonant if there are nonnegative integers $m_1, \cdots, m_n$ with $m_1 + \cdots + \ m_n \geq 2$ such that $m_1\lambda_1 + \cdots + m_n\lambda_n = \lambda_j$ for some $1 \leq j \ \leq n$. He further showed that, when resonances are present, the dynamical system can be put in a normal form

$$\dot{z} = Fz + \sum_{|m|=2}^{\infty} \Theta^m z_1^{m_1} z_2^{m_2} \cdots z_n^{m_n} \tag{12}$$

where $\Theta^m = (\Theta_1^m, \cdots, \Theta_n^m)$ is a vector constant whose j^{th}-component is zero when there is no resonance of order m associated to the eigenvalue λ_j.

Notations: For a vector field $f(x) = (f_1(x), \cdots, f_n(x))$ on $\Re^n$ and a function h in x-coordinates $= (x_1, \cdots, x_n)$, we denote by

$$\mathcal{L}_f h(x) = \frac{\partial h}{\partial x_1} f_1(x) + \frac{\partial h}{\partial x_2} f_2(x) + \cdots + \frac{\partial h}{\partial x_n} f_n(x) \tag{13}$$

the Lie derivative of h along the vector field f, and recursively, we define the Lie-derivatives

$$\mathcal{L}_f^0 h(x) = h(x), \mathcal{L}_f^j h(x) = \mathcal{L}_f \mathcal{L}_f^{j-1} h(x), j = 1,2,\ldots,\infty \tag{14}$$

For another vector field $g(x) = (g_1(x), \cdots, g_n(x))$ on $\Re^n$, we define the Lie bracket $[f, \ g]$ between the two vector fields as a new vector field

$$[f, g](x) = \left(\mathcal{L}_f g_1(x) - \mathcal{L}_g f_1(x), \cdots, \mathcal{L}_f g_n(x) - \mathcal{L}_g f_n(x)\right) \tag{15}$$

and, for simplicity, we denote such vector field as $ad_f g(x) = [f, \ g](x)$, and recursively, we define

$$ad_f^0 g(x) = g(x), ad_f^j g(x) = [f, ad_f^{j-1} g](x), j = 1,2,\ldots,\infty \tag{16}$$

Let $\Phi: \Re^n \mapsto \Re^n$be a local diffeomorphism with $\Phi(0) = 0$, giving rise to new coordinates $z = \Phi(x)$. The vector field f is transported by Φ into a new vector field, denoted $\bar{f}(z) \triangleq \Phi_* f(z)$, whose components $\bar{f}(z) = \left(\bar{f}_1(z), ..., \bar{f}_n(z)\right)$ are given for all $1 \leq j \leq n$ by

$$\bar{f}_j(z) = \mathcal{L}_f \Phi_j(\Phi^{-1}(z)) = \frac{\partial \Phi_j}{\partial x_1} f_1(\Phi^{-1}(z)) + \frac{\partial \Phi_j}{\partial x_2} f_2(\Phi^{-1}(z)) + \cdots + \frac{\partial \Phi_j}{\partial x_n} f_n(\Phi^{-1}(z)) \tag{17}$$

Below we recall the method we provided in [25] to solve the problem of straightening a vector field around a nonsingular point. Without loss of generality, we will assume the nonsingular point to be $\Re^n$.

Theorem 2: Let $v = \left(v_1, ..., \quad v_m\right)$ be aanalytic vector field on $\Re^n$and $\sigma(x) = \frac{1}{v_k(x)}$.

i. Define $z = \Phi(x)$ by its components as following

$$\begin{aligned} \Phi_j(x) &= x_j + \sum_{s=1}^{\infty} \frac{(-1)^s x_k^s}{s!} \mathcal{L}_{\sigma v}^{s-1}(\sigma v_j)(x), \quad j \neq k \\ \Phi_j(x) &= \sum_{s=1}^{\infty} \frac{(-1)^{s+1} x_k^s}{s!} \mathcal{L}_{\sigma v}^{s-1}(\sigma)(x), \quad j = k \end{aligned} \tag{18}$$

The local diffeomorphism Φ satisfies $\Phi_*(v)(z) = (0, ..., 0, 1, 0, ..., 0) \triangleq \frac{\partial}{\partial x_k}$.

ii. The local diffeomorphism $x = \Psi(z)$ whose components are given by

$$\begin{aligned} \Psi_j(z) &= z_j + \sum_{s=1}^{\infty} \frac{z_n^s}{s!} \left(\sum_{i=0}^{s-1} (-1)^i C_n^i \, \partial_{z_k}^i \mathcal{L}_v^{s-i-1}(v_j)(z) \right), \quad j \neq k \\ \Psi_j(z) &= \sum_{s=1}^{\infty} \frac{z_n^s}{s!} \left(\sum_{i=0}^{s-1} (-1)^i C_n^i \, \partial_{z_k}^i \mathcal{L}_v^{s-i-1}(v_n)(z) \right), \quad j = k \end{aligned} \tag{19}$$

is the inverse of $z = \Phi(x)$, that is, $\Phi(\Psi(z)) = z$ and $\Psi(\Phi(x)) = x$ such that $\frac{\partial \Psi}{\partial z_k} = v(\Psi(z))$.

The series proposed above are not Taylor series or series in the variable x_k (resp. z_k). Indeed, the coefficients $\mathcal{L}_{\sigma v}^{s-1}\left(\sigma v_j\right)(x)$ and $\mathcal{L}_{\sigma v}^{s-1}(\sigma)(x)$ are functions that depend on the variables x_k (resp. z_k). Above, the notation $\partial_{z_k}^i h$ means the i^{th}-derivative of h about the variable z_k. We refer to [tall-adjm] for more details and the generalization of Frobenius theorem to the straightening of a set of vector fields as stated below.

Theorem 3: Let $v^1(x), ..., v^m(x)$ be a set of analytic vector fields on $\Re^n$ such that the distribution $\mathcal{D}(x) = \left\{v^1(x), ..., v^m(x)\right\}^1$ is involutive and of maximal rank m ≤ n in a neighborhood $\mathcal{U} \subseteq \Re^n$ of the origin. There exist an open neighborhood $0 \epsilon \Omega \subseteq \mathcal{U}$ and a change of coordinates $z = \Phi(x)$ such that $\Phi_*\left(v^i\right)(z) = \frac{\partial}{\partial z_i}$ for all $z \epsilon \Phi(\Omega)$ and $i = 1, \quad ..., \quad m$.

We proposed a constructive way to find the diffeomorphism Φ through successive applications of Frobenius theorem.

3. Control systems and feedback linearization

Let us reconsider the control-affine nonlinear system with outputs

$$\Sigma : \begin{cases} \dot{x} = f(x) + g(x)u = f(x) + g_1(x)u_1 + \cdots + g_m(x)u_m, \ x\epsilon \Re^n, u \epsilon \Re^m \\ y = h(x) = (h_1(x), \cdots, h_p(x)) \end{cases} \tag{20}$$

The input-output feedback linearization as stated earlier is to find new coordinates system $z = (z_1, \cdots, z_n) \epsilon \Re^n$ and new inputs $v = (v_1, \cdots, v_m) \epsilon \Re^m$ under which the system Σ has linear dynamics and linear outputs. This problem has been connected directly to the notion of relative degree. Indeed, one needs to differentiate the outputs repeatedly until the inputs appear. Formally, if there exists $\gamma_i > 0$ such that $\mathcal{L}_{g_j}\mathcal{L}_f^k h_i(x) = 0$ for all $1 \leq j \leq m$ and $0 \leq k \leq \gamma_i - 2$ with $\mathcal{L}_{g_j}\mathcal{L}_f^{\gamma_i - 1} h_i(x) \neq 0$ for some j, we say that γ_i is the relative degree of the j^{th} output. In other words, γ_i is the smallest integer k for which the k^{th}-derivative $y_i^{(k)}$ of $y_i$depends explicitly on the input u. The set $\{\gamma_1, \cdots, \gamma_m\}$ is called *vector relative degree* associated to the outputs of Σ. It is well known that taking $z_k^i = \mathcal{L}_f^{k-1} h_i(x)$ for $1 \leq k \leq \gamma_i$ and completing the coordinates with $z_{\gamma_i + 1}^i \cdots, z_n^i$, the system can be expressed into m-subsystems of the form

$$\begin{cases} \dot{z}_1^i = z_2^i \\ \dot{z}_2^i = z_3^i \\ \vdots \\ \dot{z}_{\gamma_i - 1}^i = z_{\gamma_i}^i \\ \dot{z}_{\gamma_i}^i = f_{\gamma_i}(z) + g_{\gamma_1}^1(z)v_1 + \cdots + g_{\gamma_i}^m(z)v_m \\ \vdots \\ \dot{z}_{n_i}^i = f_{n_i}(z) + g_{n_1}^1(z)v_1 + \cdots + g_{n_i}^m(z)v_m \\ y_i = z_1^i \end{cases} \tag{21}$$

for $1 \leq i \leq m$ with $n_1 + \cdots + n_m = m$. Thus, the system becomes a connection between a linear and nonlinear systems and this has been known as partial feedback linearization. A necessary and sufficient condition for exact linearization, that is, for a multi-input multi-output system to be transformed into a chain of integrators

$$(BR)\begin{cases} \dot{z}_1^1 = z_2^1 & \dot{z}_1^m = z_2^m \\ \dot{z}_2^1 = z_3^1 & \dot{z}_2^m = z_3^m \\ \vdots & \vdots \\ \dot{z}_{\gamma_1-1}^1 = z_{\gamma_1}^i & \cdots \quad \dot{z}_{\gamma_1-1}^m = z_{\gamma_1}^m \\ \dot{z}_{\gamma_1}^1 = v_1 & \dot{z}_{\gamma_1}^m = v_m \\ y_1 = z_1^1 & y_m = z_1^m \end{cases} \tag{22}$$

is that it has a vector relative degree $\{\gamma_1, \cdots, \gamma_n\}$ such that $\gamma_1 + \cdots + \gamma_m = n$.

Obviously, different outputs will lead to different cascade systems: A system can be linearized with respect to some outputs and fail to be linearizable with respect to a different set of outputs. If we consider a control-affine system without outputs, then the linearization problem (Problem 2) is equivalent to solving a system of partial differential equation. Indeed, two affine control systems

$$\Sigma : \dot{x} = f(x) + g(x)u = f(x) + g_1(x)u_1 + \cdots + g_m(x)u_m, \ x \epsilon \Re^n, u \epsilon \Re^m \tag{23}$$

and

$$\overline{\Sigma} : \dot{z} = \overline{f}(z) + \overline{g}(z)v = \overline{f}(z) + \overline{g}_1(z)v_1 + \cdots + \overline{g}_m(z)v_m, \ z \epsilon \Re^n, v \epsilon \Re^m \tag{24}$$

are feedback equivalent via static state transformations $z = \Phi(x)$ and feedback $u = \alpha(x) + \beta(x)v$ if and only if

$$(PDEs) \ : \begin{cases} \bar{f}(\Phi(x)) = \frac{\partial \Phi}{\partial x}\big(f(x) + g(x)\alpha(x)\big) \\ \bar{g}(\Phi(x)) = \frac{\partial \Phi}{\partial x}\big(g(x)\beta(x)\big) \end{cases} \tag{25}$$

In particular, the control-affine system Σ is static state feedback equivalent to a controllable linear system if and only the system of partial differential equations

$$(PDEs) \ : \begin{cases} A\Phi(x) = \frac{\partial \Phi}{\partial x}\big(f(x) + g(x)\alpha(x)\big) \\ B = \frac{\partial \Phi}{\partial x}\big(g(x)\beta(x)\big) \end{cases} \tag{26}$$

is solvable in Φ, α, and β with Φ a diffeomorphism around the origin, and β invertible. A geometric characterization of feedback linearization was obtained by Jakubczyk and Respondek [4] and independently by Hunt and Su [5]

Theorem 4: The system Σ is feedback equivalent to a controllable linear system Λ around an equilibrium point $x_0 = 0$ if and only if the following two conditions are satisfied

http://dx.doi.org/10.5772/64689

(F1) $\dim \mathcal{D}^n(x) = n$

(F2) $[\mathcal{D}^j,\ \mathcal{D}^k] \subseteq \mathcal{D}^j$ for any $1 \le k \le j$.

Above $\mathcal{D}^j(x)$ stand for distributions defined recursively by

$$\mathcal{D}^j(x) = span\left\{g_i(x), ad_f g_i(x), \ldots, ad_f^{j-1} g_i(x), 1 \le i \le m\right\} \tag{27}$$

and $[\mathcal{D}^j,\ \mathcal{D}^k]$ as the distribution spanned by all Lie brackets of the two distributions. The first condition (F1) stands for the rank condition while the second (F2) is referred as the involutivity condition.

Thus, to find the largest linear subsystem, the outputs need not to be predefined.

In this paper, we consider only systems without outputs and look to find such largest linear subsystem. First, an affine system $\sum : \dot{x} = f(x) + g(x)u,\ x \epsilon R^n, u \epsilon R^m$ is said to be partially static state feedback linearizable if there exists a coordinate system $z = (z_1, \cdots, z_n)$ and feedback in which the system takes the form

$$\Lambda_p : \begin{cases} \frac{dz^1}{dt} &= \tilde{f}(z^1, z^2) + \tilde{g}(z^1, z^2)v \\ \frac{dz^2}{dt} &= Az^2 + Bv \end{cases} \tag{28}$$

where $z^1 = (z_1, \cdots, z_q)$ and $z^2 = \left(z_{q+1}, \cdots, z_n\right)$.

Remark 1: Notice that the form above is also equivalent to

$$\begin{cases} \frac{dz^1}{dt} &= Az^1 + Bv \\ \frac{dz^2}{dt} &= \tilde{f}\left(z^1, z^2\right) + \tilde{g}\left(z^1, z^2\right)v \end{cases} \tag{29}$$

by reordering the variables accordingly. In the sequel, we will refer more to the former form. The following result can be found in [17]

Theorem 5: Consider a control affine system Σ.

i. If Σ is locally state space equivalent at x_0 to a partially linear system Λ_p then dim $\mathcal{L}^2(x) < n$ in a neighbourhood of x_0.

ii. Assume that Σ satisfies $\dim \mathcal{L}_0(x) = n$ and that $\dim \mathcal{L}^2(x) = \rho$ in a neighbourhood of x_0. Then, Σ is locally state space equivalent at x_0 to a partially linear system Λ_p, such that

the dimension of the linear subsystem is $\dim z^2 = n - \rho$, and moreover, the linear subsystem is controllable.

We will provide a step-by-step procedure to write the system as a cascade of a nonlinear subsystem and a linear subsystem with highest dimension. Notice that a geometrical approach has been used in [14, 16] where the characterization depends on controllability indices associated to some lie algebras.

4. Algorithm for partial feedback linearization

We first consider a single-input control system

$$\Sigma : \ \dot{x} = f(x) + g(x)u, \ \ x \epsilon \Re^n, u \epsilon \Re \tag{30}$$

and we assume that its linear approximation $\dot{x} = Fx + Gu$ is controllable with $F = \frac{\partial f}{\partial x}(0)$ and $G = g(0)$. Without loss of generality, we can also assume that the pair (F, G) is in Brunovsky canonical form.

Step 0: We apply the Frobenius theorem to find coordinates $y = \varphi(x)$ that rectifies the vector field g, that is, such that $\varphi_*(g) = (0, ..., 0, 1) \triangleq b$ and transform the system as

$$\Sigma : \ \dot{y} = \tilde{f}(y) + bu, \ \ y \epsilon \Re^n, u \epsilon \Re. \tag{31}$$

Completing this step with the push-forward transformation

$$\begin{cases} z_1 = y_1 \\ \vdots \\ z_{n-1} = y_{n-1} \\ z_n = \tilde{f}_{n-1}(y) \\ v = \dfrac{\partial \tilde{f}_{n-1}}{\partial y_1} \tilde{f}_1(y) + \cdots + \dfrac{\partial \tilde{f}_{n-1}}{\partial y_n} \tilde{f}_n(y) + \dfrac{\partial \tilde{f}_{n-1}}{\partial y_n} u \end{cases} \tag{32}$$

the system is transformed as

$$\Sigma : \ \dot{z} = \overline{f}(z) + bv, \ \ z \epsilon \Re^n, v \epsilon \Re \tag{33}$$

where

http://dx.doi.org/10.5772/64689

$$\bar{f}(z)=\begin{pmatrix} \bar{f}_1(z_1,\cdots,z_n) \\ \bar{f}_2(z_1,\cdots,z_n) \\ \vdots \\ \bar{f}_{n-2}(z_1,\cdots,z_n) \\ z_n \\ 0 \end{pmatrix} \tag{34}$$

Step 1: We reset the original notation, that is, replace the variable z by x, and $\bar{f}(z)$ by $f(x)$. Then, we decompose $f(x)$ as following

$$f(x)=\begin{pmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_{n-2}(x) \\ x_n \\ 0 \end{pmatrix}=\begin{pmatrix} f_1(x_1,\cdots,x_{n-1}) \\ f_2(x_1,\cdots,x_{n-1}) \\ \vdots \\ f_{n-2}(x_1,\cdots,x_{n-1}) \\ 0 \\ 0 \end{pmatrix}+x_n\begin{pmatrix} g_1(x_1,\cdots,x_{n-1}) \\ g_2(x_1,\cdots,x_{n-1}) \\ \vdots \\ g_{n-2}(x_1,\cdots,x_{n-1}) \\ 1 \\ 0 \end{pmatrix}+x_n^2\begin{pmatrix} G_1^n(x) \\ G_2^n(x) \\ \vdots \\ G_{n-2}^n(x) \\ 0 \\ 0 \end{pmatrix} \tag{35}$$

If $G_{n-2}^n(x_1,\cdots,x_n)\neq 0$, then the algorithm stops. This means that the dimension of the largest linear subsystem is 2. In case $G_{n-2}^n(x_1,\cdots,x_n)=0$, we define ρ_n the largest j such that $G_j^n(x_1,\cdots,x_n)\neq 0$. If $G_j^n(x_1,\cdots,x_n)=0$ for all $1\leq j\leq n-2$, then we put $\rho_n=0$. We then apply the Frobenius theorem to straighten the vector field

$$g(x)=\begin{pmatrix} g_1(x_1,\cdots,x_{n-1}) \\ g_2(x_1,\cdots,x_{n-1}) \\ \vdots \\ g_{n-2}(x_1,\cdots,x_{n-1}) \\ 1 \\ 0 \end{pmatrix} \tag{36}$$

by defining coordinates $y=\varphi(x)$ such that $\varphi_*(g)=(0,...,0,1,0)\triangleq Ab$. Notice that, because g depends only on the variables $x_1,\cdots,x_{n-1}$, so do the first (n–1) components of the diffeomorphism ϕ. Thus, the system is transformed as

$$\Sigma: \quad \dot{y} = \underbrace{\begin{pmatrix} f_1(y_1,\cdots,y_{n-1}) \\ f_2(y_1,\cdots,y_{n-1}) \\ \vdots \\ f_{n-2}(y_1,\cdots,y_{n-1}) \\ 0 \\ 0 \end{pmatrix} + y_n \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \end{pmatrix} + y_n^2 \begin{pmatrix} G_1^n(y_1,\cdots,y_n) \\ G_2^n(y_1,\cdots,y_n) \\ \vdots \\ G_{n-2}^n(y_1,\cdots,y_n) \\ 0 \\ 0 \end{pmatrix}}_{\tilde{f}(y)} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{pmatrix} u, \tag{37}$$

We thus apply the push-forward transformation

$$\begin{cases} z_1 = y_1 \\ \vdots \\ z_{n-2} = y_{n-2} \\ z_{n-1} = f_{n-2}(y_1,\cdots,y_{n-1}) \\ z_n = \dfrac{\partial z_{n-1}}{\partial y_1}\tilde{f}_1(y) + \cdots + \dfrac{\partial z_{n-1}}{\partial y_{n-2}}\tilde{f}_{n-2}(y) + \dfrac{\partial z_{n-1}}{\partial y_{n-1}} y_n \\ v = \dfrac{\partial z_n}{\partial y_1}\tilde{f}_1(y) + \cdots + \dfrac{\partial z_n}{\partial y_{n-1}}\tilde{f}_{n-1}(y) + \dfrac{\partial z_n}{\partial y_n} u \end{cases} \tag{38}$$

to bring the system into the form

$$\Sigma: \quad \dot{z} = \underbrace{\begin{pmatrix} f_1(z_1,\cdots,z_{n-1}) \\ f_2(z_1,\cdots,z_{n-1}) \\ \vdots \\ \vdots \\ f_{n-3}(z_1,\cdots,z_{n-1}) \\ z_{n-1} \\ 0 \\ 0 \end{pmatrix} + z_n \begin{pmatrix} F_1^n(z_1,\cdots,z_n) \\ F_2^n(z_1,\cdots,z_n) \\ \vdots \\ F_{\rho_n}^n(z_1,\cdots,z_n) \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}}_{} + z_n \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v, \quad z \in \Re^n, v \in \Re \tag{39}$$

or in much compact form

$$\Sigma: \quad \dot{z} = f(z_1,\cdots,z_{n-1}) + z_n F^n(z) + ABz_n + Bu, \quad z \in \Re^n, u \in \Re^m \tag{40}$$

with $\rho_n = \max\left\{j, 1 \le j \le n-2, \left|F_j^n(z_1,\cdots,z_n) \neq 0\right.\right\}$. Moreover, and more importantly, we also have

$$F^n(0) = 0 \quad \text{and} \quad \frac{\partial F_{\rho_n}^n}{\partial z_n} \neq 0. \tag{41}$$

http://dx.doi.org/10.5772/64689

Remark 2

1. Please notice that the vector field $z_n F^n(z)$ contains all nonlinearities including terms that are linear in z_n but whose coefficient depends on the variables $z_1, ..., z_{n-1}$.

2. The Frobeinus theorem applied to the vector field g could have been restricted by taking the first ρ_n components of vector field g equal zero. This is due to the fact that, by applying the push-forward transformation above, we regenerate those terms as y_n depends on all variables $z_1, ..., z_n$.

Step 2: We reset the original notation, that is, replace the variable z by x. Then, we decompose $f(x)$ as following

$$f(x) = \begin{pmatrix} f_1(x_1,\cdots,x_{n-1}) \\ f_2(x_1,\cdots,x_{n-1}) \\ \vdots \\ f_{n-3}(x_1,\cdots,x_{n-1}) \\ x_{n-1} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} f_1(x_1,\cdots,x_{n-2}) \\ f_2(x_1,\cdots,x_{n-2}) \\ \vdots \\ f_{n-3}(x_1,\cdots,x_{n-2}) \\ 0 \\ 0 \\ 0 \end{pmatrix} + x_{n-1} \begin{pmatrix} g_1(x_1,\cdots,x_{n-2}) \\ g_2(x_1,\cdots,x_{n-2}) \\ \vdots \\ g_{n-3}(x_1,\cdots,x_{n-2}) \\ 1 \\ 0 \\ 0 \end{pmatrix} + x_{n-1}^2 \begin{pmatrix} G_1\ (x_1,\cdots,x_{n-1}) \\ G_2^n(x_1,\cdots,x_{n-1}) \\ \vdots \\ G_{n-3}^n(x_1,\cdots,x_{n-1}) \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{42}$$

If $G_{n-3}^n(x_1,\cdots,x_n) \neq 0$, then the dimension of the largest linear subsystem is less or equal to 3. We denote by ρ_{n-1} the largest j such that $G_j^n(x_1,\cdots,x_n) \neq 0$. If $G_j^n(x_1,\cdots,x_n) = 0$ for all $1 \leq j \leq n-3$, then we put $\rho_{n-1} = 0$. We define $\bar{\rho}_{n-1} = \max\{\rho_{n-1},\ \rho_n\}$ as the updated largest component that cannot be cancelled or, equivalently, such that the dimension of the largest linear subsystem is less or equal to $n - \bar{\rho}_{n-1}$.

We then apply the Frobenius theorem to straighten the vector field

$$g(x) = \begin{pmatrix} g_1(x_1,\cdots,x_{n-2}) \\ g_2(x_1,\cdots,x_{n-2}) \\ \vdots \\ g_{n-3}(x_1,\cdots,x_{n-2}) \\ 1 \\ 0 \\ 0 \end{pmatrix} \tag{43}$$

by defining coordinates $y = \varphi(x)$ such that $\varphi_*(g) = (0, ..., 0, 1, 0, 0) \triangleq A^2 b$. Notice that, because g depends only on the variables $x_1, \cdots, x_{n-2}$, so do the first $(n-2)$ components of the diffeomorphism ϕ. Thus, the system is transformed as

$$\Sigma : \dot{z} = \begin{pmatrix} f_1(z_1, \cdots, z_{n-2}) \\ f_2(z_1, \cdots, z_{n-2}) \\ \vdots \\ \vdots \\ f_{n-3}(z_1, \cdots, z_{n-2}) \\ z_{n-2} \\ 0 \\ 0 \\ 0 \end{pmatrix} + z_{n-1} \begin{pmatrix} F_1^{n-1}(z_1, \cdots, z_{n-1}) \\ F_2^{n-1}(z_1, \cdots, z_{n-1}) \\ \vdots \\ F_{\bar{\rho}_{n-1}}^{n-1}(z_1, \cdots, z_{n-1}) \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} + z_n \begin{pmatrix} F_1^n(z_1, \cdots, z_n) \\ F_2^n(z_1, \cdots, z_n) \\ \vdots \\ F_{\bar{\rho}_{n-1}}^n(z_1, \cdots, z_n) \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} + \tag{44}$$

$$z_{n-1} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + z_n \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v, \quad z \in \Re^n, v \in \Re$$

We thus apply the push-forward transformation

$$\begin{cases} z_1 = y_1 \\ \vdots \\ z_{n-3} = y_{n-3} \\ z_{n-2} = f_{n-3}(y_1, \cdots, y_{n-2}) \\ z_{n-1} = \dfrac{\partial z_{n-2}}{\partial y_1} \tilde{f}_1(y) + \cdots + \dfrac{\partial z_{n-2}}{\partial y_{n-3}} \tilde{f}_{n-3}(y) + \dfrac{\partial z_{n-2}}{\partial y_{n-2}} y_{n-1} \\ z_n = \dfrac{\partial z_{n-1}}{\partial y_1} \tilde{f}_1(y) + \cdots + \dfrac{\partial z_{n-1}}{\partial y_{n-2}} \tilde{f}_{n-2}(y) + \dfrac{\partial z_{n-1}}{\partial y_{n-1}} y_n \\ v = \dfrac{\partial z_n}{\partial y_1} \tilde{f}_1(y) + \cdots + \dfrac{\partial z_n}{\partial y_{n-1}} \tilde{f}_{n-1}(y) + \dfrac{\partial z_n}{\partial y_n} u \end{cases} \tag{45}$$

to bring the system into the form

http://dx.doi.org/10.5772/64689

$$\Sigma:\ \dot{z}=\begin{pmatrix} f_1(z_1,\cdots,z_{n-2}) \\ f_2(z_1,\cdots,z_{n-2}) \\ \vdots \\ \vdots \\ f_{n-3}(z_1,\cdots,z_{n-2}) \\ z_{n-2} \\ 0 \\ 0 \\ 0 \end{pmatrix} + z_{n-1}\begin{pmatrix} F_1^{n-1}(z_1,\cdots,z_{n-1}) \\ F_2^{n-1}(z_1,\cdots,z_{n-1}) \\ \vdots \\ F_{\overline{\rho}_{n-1}}^{n-1}(z_1,\cdots,z_{n-1}) \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} + z_n\begin{pmatrix} F_1^{n}(z_1,\cdots,z_{n}) \\ F_2^{n}(z_1,\cdots,z_{n}) \\ \vdots \\ F_{\overline{\rho}_{n-1}}^{n}(z_1,\cdots,z_{n}) \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} + \tag{46}$$

$$z_{n-1}\begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + z_n\begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} v,\ \ z \in \Re^n, v \in \Re$$

or in much compact form

$$\Sigma:\ \ \dot{z}=f(z_1,\cdots,z_{n-2})+z_{n-1}F^{n-1}(z_1,\cdots,z_{n-1})+z_nF^n(z)+A^2bz_{n-1}+Abz_n+bu, \tag{47}$$

with $F^{n-1}(0)=F^n(0)=0$ and either $\dfrac{\partial F_{\overline{\rho}n-1}^{n}}{\partial z_n}\neq 0$ or $\dfrac{\partial F_{\overline{\rho}n-1}^{n-1}}{\partial z_{n-1}}\neq 0$.

General step: Let us assume that the system has been transformed such that it takes the form

$$\Sigma:\ \ \dot{x}=f(x_1,\cdots,x_k)+\sum_{i=k}^{n-1}\left(x_{i+1}F^{i+1}(x_1,\cdots,x_{i+1})+A^{n-i}Bx_{i+1}\right)+Bu,\ \ x\in\Re^n, u\in\Re \tag{48}$$

where $F^{i+1}(0)=0$ for all $k\le i\le n-1$ and $\dfrac{\partial F_{\rho}^{i+1}}{\partial z_{i+1}}\neq 0$ for some $i,\quad k\le i\le n-1$ with ρ being the largest nonzero component among those of the vector fields $F^{k+1},\ldots,F^n$. We will write

$$f\left(x_1,\cdots,x_k\right)=\begin{pmatrix} f_1\left(x_1,\cdots,x_k\right) \\ f_2\left(x_1,\cdots,x_k\right) \\ \vdots \\ f_{k-2}\left(x_1,\cdots,x_k\right) \\ x_k \\ 0 \\ \vdots \\ 0 \end{pmatrix}\ \text{and}\ F^{i+1}\left(x\right)=\begin{pmatrix} F_1^{i+1}\left(x_1,\cdots,x_{i+1}\right) \\ F_2^{i+1}\left(x_1,\cdots,x_{i+1}\right) \\ \vdots \\ F_{\rho^*}^{i+1}\left(x_1,\cdots,x_{i+1}\right) \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{49}$$

Then, we decompose the vector field f as follows

$$f(x)=\begin{pmatrix} f_1(x_1,\cdots,x_{k-1}) \\ f_2(x_1,\cdots,x_{k-1}) \\ \vdots \\ f_{k-2^*}(x_1,\cdots,x_{k-1}) \\ 0 \\ \vdots \\ 0 \end{pmatrix}+x_k\begin{pmatrix} g_1(x_1,\cdots,x_{k-1}) \\ g_2(x_1,\cdots,x_{k-1}) \\ \vdots \\ g_{k-2}(x_1,\cdots,x_{k-1}) \\ 1 \\ \vdots \\ 0 \end{pmatrix}+x_k^2\begin{pmatrix} G_1(x_1,\cdots,x_k) \\ G_2(x_1,\cdots,x_k) \\ \vdots \\ G_{k-2}(x_1,\cdots,x_k) \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{50}$$

If the largest nonzero component of the vector field $G(x)$ is less or equal to ρ, then move to the next step. If that largest component is greater than ρ, then update ρ as this component and apply Frobenius theorem to straighten the vector field $g(x)$ and follow by a push-forward transformation. Any time in the process the value of $\rho = n - 2$, the algorithm will stop; if not until, we reach the last step.

5. Examples

In this section, we consider few examples to illustrate the partial feedback linearization algorithm.

Example 1: Consider a simplified model of a VTOL with dynamics [29] (see **Figure 1**).

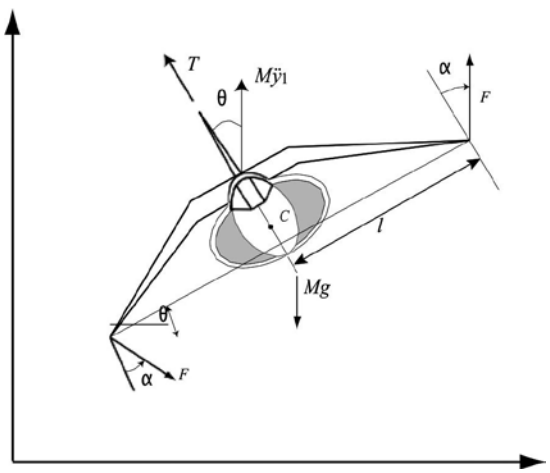

Figure 1. Forces acting on a VTOL aircraft.

http://dx.doi.org/10.5772/64689

$$\begin{cases} \ddot{x} = -\sin(\theta)\frac{T}{M} + \cos(\theta)\frac{2\sin(\alpha)}{M}F \\ \ddot{y} = -\cos(\theta)\frac{T}{M} + \sin(\theta)\frac{2\sin(\alpha)}{M}F - g \\ \ddot{\theta} = \frac{2l}{J}\cos(\alpha)F \end{cases} \tag{51}$$

where M, J, l, and g denote the mass, moment of inertia, distance between wingtips and gravitational acceleration. The control inputs are the thrust T, and the rolling moment due to the torque F, whose direction forms a fixed angle α with the horizontal body axis. The position of center mass and the roll angle with respect to the horizon are (x,y), and θ, while $(\dot{x}, \dot{y})$ and $\dot{\theta}$ stand for their respective velocities.

Let $x_1 = x, \quad x_2 = \dot{x}, \quad x_3 = \theta, \quad x_4 = \dot{\theta}, \quad x_5 = y, \quad x_6 = \dot{y}$ with control inputs

$$u_1 = \frac{2lF}{J}\cos\alpha \tag{52}$$

and

$$u_2 = -\sin(\theta)\frac{T}{M} + \cos(\theta)\frac{2\sin(\alpha)}{M}F - g \tag{53}$$

The system rewrites in the form

$$\Sigma: \ \dot{x} = f(x) + g_1(x)u_1 + g_2(x)u_2, x = (x_1, \ldots, x_6) \in \Re^6 \tag{54}$$

with

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} x_2 \\ g\tan x_3 \\ x_4 \\ 0 \\ x_6 \\ 0 \end{pmatrix}, \ g_1(x) = \begin{pmatrix} 0 \\ \eta(x_3) \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \text{ and } g_2(x) = \begin{pmatrix} 0 \\ \tan x_3 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \tag{55}$$

where

$$\eta\left(x_3\right) = \frac{J\tan\alpha}{Ml}\left(\frac{cos^2 x_3 - sin^2 x_3}{\cos x_3}\right) \tag{56}$$

We showed in [25] that the change of coordinates

$$z = \varphi(x) \triangleq \begin{cases} z_1 = x_1 \\ z_2 = x_2 - x_4\eta(x_3) - x_6 \tan x_3 \\ z_3 = x_3 \\ z_4 = x_4 \\ z_5 = x_5 \\ z_6 = x_6 \end{cases} \tag{57}$$

transformed the system into $\bar{\Sigma}: \dot{z} = \bar{f}(z) + \bar{g}_1(z)u_1 + \bar{g}_2(z)u_2, z = (z_1, \ldots, z_6) \epsilon \Re^6$ where

$$\bar{\boldsymbol{f}}(z) = \begin{pmatrix} z_2 + z_4\eta(z_3) + z_6 \tan z_3 \\ \text{gtan } z_3 - \eta'(z_3)z_4^2 - z_6 z_4 sec^2(z_3) \\ z_4 \\ 0 \\ z_6 \\ 0 \end{pmatrix}, \ \bar{g}_1(z) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \text{ and } \bar{g}_2(z) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \tag{58}$$

The distribution generated by g_1 and g_2 is involutive and constant. A simple feedback

$$v_1 = -\boldsymbol{x}_1 - 2\boldsymbol{x}_3 + 2\boldsymbol{x}_5^2\boldsymbol{x}_6 - 2\boldsymbol{x}_6^2 + \boldsymbol{u}_1 + \boldsymbol{u}_2 \text{ and } v_2 = -\boldsymbol{x}_4 - \boldsymbol{x}_5 + \boldsymbol{x}_4\boldsymbol{x}_5 + \boldsymbol{u}_2 \tag{59}$$

transforms the system so as

$$f(x) = \begin{pmatrix} x_1 + x_2 - x_4^2 + 2x_4x_5 \\ x_3 - x_6^2 \\ 0 \\ x_5 \\ 0 \\ x_4 + x_5^2 - x_6 \end{pmatrix}, \ g_1(x) = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ and } g_2(x) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \tag{60}$$

We then decompose the vector field f as

$$f(x) = \begin{pmatrix} x_1 + x_2 - x_4^2 + 2x_4x_5 \\ x_3 - x_6^2 \\ 0 \\ x_5 \\ 0 \\ x_4 + x_5^2 - x_6 \end{pmatrix} = x_3 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + x_5 \begin{pmatrix} 2x_4 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} x_1 + x_2 - x_4^2 \\ -x_6^2 \\ 0 \\ 0 \\ 0 \\ x_4 + x_5^2 - x_6 \end{pmatrix} \tag{61}$$

Here, we rectify the two vector fields (affine in x_3 and x_5) and find the change of coordinates

http://dx.doi.org/10.5772/64689

$$\begin{cases} y_1 = x_1 - x_4^2 \\ y_2 = x_2 \\ y_3 = x_3 \\ y_4 = x_4 \\ y_5 = x_5 \\ y_6 = x_6 \end{cases} \tag{62}$$

to transform the system into

$$\begin{cases} \dot{y}_1 = y_1 + y_2 \\ \dot{y}_2 = y_3 - y_6^2 \\ \dot{y}_3 = u_1 \\ \dot{y}_4 = y_5 \\ \dot{y}_5 = u_5 \\ \dot{y}_6 = y_4 - y_6 + y_5^2 \end{cases} \tag{63}$$

If we apply the push-forward transformation given by $z_3 = x_3 - x_6^2, \quad z_j = x_j, \quad j \neq 3$, and the feedback $v_1 = u_1 - 2x_6\left(x_4 + x_5^2 - x_6\right), \quad v_2 = u_2$, we take the system into

$$\Sigma : \begin{cases} \dot{z}_1 = z_1 + z_2 \\ \dot{z}_2 = z_3 \\ \dot{z}_3 = v_1 \\ \dot{z}_4 = z_5 \\ \dot{z}_5 = v_2 \\ \dot{z}_6 = z_4 - z_6 + z_5^2 \end{cases} \tag{64}$$

with $\rho = 4$ being the dimension of the largest linear subsystem.

Author details

Issa Amadou Tall

Address all correspondence to: tallia@elac.edu

East Los Angeles College Avenida Cesar Chavez, Monterey Park, CA, USA

References

[1] P. Brunovsky. A classification of linear controllable systems. Kybernetica. 1970;3(6): 173–187.

[2] A. J. Krener. On the equivalence of control systems and the linearization of nonlinear systems. SIAM Journal on Control. 1973;11:670–676.

[3] R. W. Brockett. Feedback invariants for nonlinear systems. Proceedings of 7th IFAC Congress, Helsinki. 1978;:1115–1120.

[4] B. Jakubczyk and W. Respondek. On linearization of control systems. Bulletin Academie Polonaise des Sciences Series Mathematics. 1980;28:517–522.

[5] L. R. Hunt and R. Su. Linear equivalents of nonlinear time varying systems. In: Proceedings of Mathematical Theory of Networks & Systems; August 5-7; Santa Monica, CA. USA:1981. p. 119–123.

[6] B. Charlet, J. Levine and R. Marino. Dynamic feedback linearization, SIAM Journal on Control Optimization. 1991;(29):38–57.

[7] B. Charlet, J. Levine and R. Marino. Sufficient conditions for dynamic state feedback linearization. Systems & Control Letters. 1989;(13):143–151.

[8] Zhendong Sun and S. S. Ge, Nonregular feedback linearization: a nonsmooth approach, in IEEE Transactions on Automatic Control, vol. 48, no. 10, pp. 1772–1776, Oct. 2003.

[9] S-J. Lie and W. Respondek. Orbital feedback linearization of multi-input control systems. International Journal of Robust and Nonlinear Control. 2015;25(1):1352–1378.

[10] C. Nielsen and M. Maggiore. On local transverse feedback linearization. SIAM Journal of Control and Optimization. 2008;47(5):2227–2250.

[11] A. Isidori and A. J. Krener. On feedback equivalence of nonlinear systems. Systems & Control Letters. 1982;2(2):118–121.

[12] A. Isidori, C. Gori-Giorgi, A. J. Krener, and S. Monaco. Nonlinear decoupling via feedback: A differential geometric approach. IEEE Transactions on Automatic Control. 1982;26(2):331–345.

[13] L. R. Hunt, R. Su, and G. Meyer. Design for multi-input nonlinear systems. In: R. W. Brockett, R. S. Milman, and H. Sussmann, editors. Differential Geometric Control Theory. Boston, USA: Birkhauser; 1983. p. 268–298.

[14] R. Marino. On the largest feedback linearizable subsystem. Systems & Control Letters. 1986; 6(5):345–351.

[15] A. J. Krener, A. Isidori, W. Respondek. Partial and robust linearization by feedback. In: 22nd IEEE Conference on Decision and Control; December 14-16; San Antonio, Texas. 1983. p. 126–130.

[16] Z. Xu and L. R. Hunt. On the largest input-output linearizable subsystem. IEEE Transactions on Automatic Control. 1996;41(1):128–132.

[17] W. Respondek. Partial linearization, decompositions and fibre linear systems. In: C. I. Byrnes, A. Lindquist, editors. Theory and Applications of Nonlinear Control Systems. Amsterdam and New York: North-Holland:1986. p. 137–154.

[18] M. W. Spong. Partial feedback linearization of underacutated mechanical systems. In: IROS94 (International Conference on Intelligent Robots and Systems); September 12-14; Munich. Germany:1994. p. 314–321.

[19] K. Pathak, J. Franch, and S. K. Agrawal. Velocity and position of wheeled inverted pendulum by partial feedback linearization. IEEE Transactions on Automatic Control. 2005;21(3):505–513.

[20] Z. Xu and L. R. Hunt. On the largest input-output linearizable subsystem. IEEE Transactions on Automatic Control. 1996;41(1):128–132.

[21] Ph. Mullhaupt. Quotient submanifolds for static feedback linearization. Systems & Control Letters. 2006;55:549–557.

[22] I. A. Tall. State and feedback linearizations of single-input control systems. Systems & Control Letters. 2010;59(7):429–441.

[23] I. A. Tall. State linearization of control systems: an explicit algorithm. In: Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference ; December 15-18; Shanghai. China:2009. p. 7448–7453.

[24] I. A. Tall. Explicit feedback linearization of control systems. In: Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference; December 15-18; Shanghai. China:2009. p. 7554–7459.

[25] I. A. Tall. Flow-Box Theorem and Beyond. African Diaspora Journal of Mathematics. 2011;11(1):75–102. DOI: www.math-res-pub.org/adjm

[26] I. A. Tall. Explicit state linearization of multi-input control systems: . In: 49th IEEE Conference on Decision and Control; December 15-17; Atlanta, GA. USA:2010. p. 7075–7080.

[27] I. A. Tall. Multi-input control systems: explicit feedback linearization. In: 49th IEEE Conference on Decision and Control; December 15-17; Atlanta, GA. USA:2010. p. 5378–5383.

[28] I. A. Tall. A new approach to exact and partial feedback linearizations. Journal of Robust and Nonlinear Control. 2012;1(1):1–32.

[29] A. Serrani, A. Isidori, C. I. Byrnes, and L. Marconi. Recent advances in output regulation of nonlinear systems. In: A. Isidori, F. Lamnabhi-Lagarrigue, and W. Respondek, editors. Nonlinear Control in the Year 2000. 2nd ed. Paris, France: Springer LNCIS; 2000. p. 409–419.

Chapter 3

Analyzing Quantum Time-Dependent Singular Potential Systems in One Dimension

Salah Menouar and Jeong Ryeol Choi

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64007

Abstract

Quantum states of a particle subjected to time-dependent singular potentials in one-dimension are investigated using invariant operator method and the Nikiforov-Uvarov method. We consider the case that the system is governed by two singular potentials which are the Coulomb potential and the inverse quadratic potential. An invariant operator that is a function of time has been constructed via a fundamental mechanics. This invariant operator is transformed to a simple one using a unitary operator, which is a time-independent invariant operator. By solving the Schrödinger equation in the transformed system, analytical forms of exact eigenvalues and eigenfunctions of the invariant operator are evaluated in a simple elegant manner with the help of the Nikiforov-Uvarov method. Eventually, the full wave functions in the original system (untransformed system) are obtained through an inverse unitary transformation from the wave functions in the transformed system. Quantum characteristics of the system associated with the wave functions are addressed in detail.

Keywords: time-dependent Hamiltonian systems, singular potentials, unitary transformation, wave function, Schrödinger equation

1. Introduction

After a seminal work of Lewis [1] for a quantum time-dependent harmonic oscillator, much attention has been paid to investigating quantum properties of time-dependent Hamiltonian systems (TDHSs). Any type of the time-dependent harmonic oscillator is a good example of TDHSs, and the study of its analytical quantum solutions requires particular mathematical techniques. The research topic of TDHSs has been gradually extended to more complicated systems beyond the one-dimensional time-dependent harmonic oscillators which are relatively

simple. The analytical forms of quantum wave functions of the time-dependent coupled oscillators have been reported by several researchers [2–4]. The system associated with a class of time-dependent singular potentials was investigated [5–10] and some of the corresponding results were applied to study the problem of a two-ion trap within a binding potential [7]. A TDHS that is described by a Hamiltonian that involves $(1/x)p + p(1/x)$ term, which in fact is necessary for the description of radial equation for a central force system, was also studied [11–13].

In this chapter, quantum features of a time-dependent singular potential system [14] will be investigated. The singular potentials that will be considered here are the combination of the inverse quadratic potential and the Coulomb potential. Singular potentials not only can be applied for describing many actual physical systems but can also serve as mathematical models for quantum field theory and elementary particle theory. The research interest for singular potentials was first shown in a context of relativistic mechanics. Various applications of the singular potentials include interatomic or intermolecular descriptions of a molecular force, the scattering problem of elementary particles, and the interaction of relativistic particles such as quark-antiquark bound states [14–16].

It was reported by Plesset [17] that there is a difficulty in the derivation of a physically accepted solution for a relativistic Coulomb-like singular potential. To overcome such difficulty, the invariant operator method together with a unitary transformation method will be used in this chapter. These methods are useful for investigating the mechanics of TDHSs, like the case that will be represented here. For a TDHS, the eigenstates of the invariant operator are the same as the Schrödinger solutions of the system when we neglect the phase factors of the wave functions [18]. The unitary transformation with a suitable operator allows us to manage a certain complicated system in a transformed space that requires relatively simple mathematical treatments for the system.

2. Singular potential system

Let us consider a one-dimensional quantum system that is described by a time-dependent Hamiltonian of the form

$$H(t) = \frac{1}{2\mu(t)}\left(p^2 + \frac{f_0}{x^2}\right) - \frac{Z(t)}{x}, \tag{1}$$

where x is the position operator and $p = -i\hbar\partial/\partial x$, $\mu(t)$, and $Z(t)$ are time-dependent coefficients with $Z(t) > 0$, and f_0 is a constant. This system is defined in the half space $x \geq 0$. The system described by Eq. (1) is different from that in Ref. [9], and a particular case of this type of Hamiltonian system can be found in Ref. [10]. The quantum problem of this Hamiltonian system is very difficult due to the explicit time dependence of parameters, and we are not

http://dx.doi.org/10.5772/64007

always possible to derive exact quantum solutions. We will find the condition for solvability of this quantum system in the subsequent development.

As is well known, a useful method for a quantum mechanical treatment of the system in the situation where there exist time-dependent parameters is to use an invariant operator method [1, 18]. An invariant of the system that is described by a time-dependent Hamiltonian $H(t)$ is constructed from the Liouville-von Neumann equation of the form

$$\frac{dI}{dt} = \frac{\partial I}{\partial t} + \frac{1}{i\hbar}[I, H] = 0. \tag{2}$$

As represented in this equation, the whole time derivative of the invariant operator I should be zero because of its definition. Let us suppose that the exact invariant has the form

$$I(x, p, t) = \alpha(t)x^2 + \gamma(t)\left(p^2 + \frac{f_0}{x^2}\right) + \beta(t)(xp + px) - \frac{\eta(t)}{x}, \tag{3}$$

where $\alpha(t)$, $\gamma(t)$, $\beta(t)$, and $\eta(t)$ are time-dependent coefficients that will be derived afterward [see Eqs. (5)–(8)]. In the case of the counterpart classical system, x and p are no longer operators, and as a consequence the expression $xp + px$ given in Eq. (3) can be reduced to $2xp$.

The substitution of Eqs. (1) and (3) into the Liouville-von Neumann equation represented in Eq. (2) gives the following equations for the coefficients:

$$\begin{aligned} &\dot{\alpha}(t) = 0, \quad \dot{\beta}(t) = -\alpha(t)/\mu(t), \quad \dot{\gamma}(t) = -2\beta(t)/\mu(t), \\ &\dot{\eta}(t) = -2\beta(t)Z(t), \quad \gamma(t)Z(t) = \eta(t)/[2\mu(t)]. \end{aligned} \tag{4}$$

By solving these equations, it is possible to determine the time-dependent coefficients. Hence, as a result of a minor evaluation, we have

$$\alpha(t) = \alpha_0, \tag{5}$$

$$\beta(t) = \beta_0 - \alpha_0 \int_0^t \frac{1}{\mu(t')} dt', \tag{6}$$

$$\gamma(t) = \gamma_0 - 2F(t), \tag{7}$$

$$\eta(t) = \frac{\eta_0}{\gamma_0^{1/2}}[\gamma_0 - 2F(t)]^{1/2}, \tag{8}$$

where

$$F(t) = \beta_0 \int_0^t \frac{1}{\mu(t')} dt' - \alpha_0 \int_0^t \left[\frac{1}{\mu(t')} \int_0^{t'} \frac{1}{\mu(t'')} dt'' \right] dt', \tag{9}$$

with an auxiliary condition for the solvability of the system, which is that the time dependence of $Z(t)$ is chosen in a way that

$$Z(t) = \frac{\eta_0}{2\gamma_0^{1/2}\mu(t)} [\gamma_0 - 2F(t)]^{-1/2}. \tag{10}$$

Now, notice that Eq. (3), with the coefficients given in Eqs. (5–8), is the exact invariant operator. If we express the eigenvalue equation of the invariant operator as $I(t)\varphi_n(t) = E_n\varphi_n(t)$, the eigenvalues E_n are time constants, due to the invariant operator not varying with time. Then we can specify the eigenstates $\varphi_n(t)$ for the operators $I(t)$ for overall range of time t.

By denoting the wave functions as $\psi_n(t)$, the Schrödinger equation is expressed in the form $i\hbar\partial\psi_n(t)/\partial t = H(t)\psi_n(t)$. For the TDHS, the wave functions are represented in terms of the eigenstates of the invariant operator, such that $\psi_n(t) = \exp[i\theta_n(t)]\varphi_n(t)$ where $\theta_n(t)$ are global phases.

Considering the Schrödinger equation, we can easily verify that $\theta_n(t)$ satisfy the relation [18]:

$$\frac{d\theta_n(t)}{dt} = \langle \varphi_n(t) | \left(i\frac{\partial}{\partial t} - \frac{H}{\hbar} \right) | \varphi_n(t) \rangle. \tag{11}$$

Hence, if the eigenstates of the invariant operator, $\varphi_n(t)$, are completely known, the corresponding global phases $\theta_n(t)$ are easily obtained by solving Eq. (11). Concerning this quantum formulation of the system based on the invariant operator, the solvability of $\psi_n(t)$ for a TDHS is noticeable.

The strategy of our manipulation for deriving exact quantum solutions of the system is that we transform the operator $I(t)$ into a simple form I_0 which is not a function of time. Then, it is easy to derive the eigenstates of I_0 associated with the transformed system because I_0 does not depend on time. The corresponding quantum results in the transformed system will be inversely transformed to the original system (untransformed system). This may lead to derive exact eigenfunctions in the original system.

For this purpose, let us first perform a unitary transformation of the eigenstates such that

http://dx.doi.org/10.5772/64007

$$\Phi_n(x) = U(x,p,t)\varphi_n(x,t), \tag{12}$$

where U is a time-dependent unitary operator given by [8]:

$$U(x,p,t) = \exp\left(\frac{i\beta(t)}{2\hbar\gamma_0}x^2\right)\exp\left[\frac{i}{2\hbar}\ln\left(\frac{\gamma(t)}{\gamma_0}\right)^{1/2}(xp+px)\right]. \tag{13}$$

The transformation of the invariant operator using this operator can be performed in a straightforward way with Eq. (3), $I_0 = UIU^{-1}$, leading to

$$I_0(x,p) = \gamma_0\left(p^2 + \frac{f_0}{x^2}\right) + \frac{\alpha_0\gamma_0 - \beta_0^2}{\gamma_0}x^2 - \frac{\eta_0}{x}. \tag{14}$$

Here, the transformed invariant operator I_0 does not depend on time as expected. Through this procedure, we can represent the eigenvalue equation for the transformed invariant operator as

$$\left[\gamma_0\left(p^2 + \frac{f_0}{x^2}\right) + \frac{\alpha_0\gamma_0 - \beta_0^2}{\gamma_0}x^2 - \frac{\eta_0}{x}\right]\Phi_n(x) = E_n\Phi_n(x). \tag{15}$$

If we put $\omega_0 = \alpha_0\gamma_0 - \beta_0^2$, it is possible to analyze the system in three cases which are $\omega_0 > 0$, $\omega_0 < 0$, and $\omega_0 = 0$. Among them, the only solvable case is the third one. Hence, let us see the system with $\omega_0 = 0$ from now on. In this case, the invariant quantity reduces to

$$I_0 = \gamma_0\left(p^2 + \frac{f_0}{x^2}\right) - \frac{\eta_0}{x}. \tag{16}$$

Then, the eigenvalue equation given in Eq. (15) becomes

$$\frac{d^2\Phi_n(x)}{dx^2} - \left(\frac{-a^2x + \nu(\nu+1)}{x^2} + \kappa_n^2\right)\Phi_n(x) = 0, \tag{17}$$

where

$$\frac{E_n}{\gamma_0\hbar^2} = -\kappa_n^2\,, \quad \frac{\eta_0}{\gamma_0\hbar^2} = a^2\,, \quad \frac{f_0}{\hbar^2} = \nu(\nu+1)\,, \tag{18}$$

with the condition that $E_n < 0$.

3. Spectrum of quantized solutions

In this section, we consider the solvable case that $\omega_0 = 0$. To evaluate the differential equation given in Eq. (17), we will use the Nikiforov-Uvarov (NU) method [19, 20] that is introduced in Appendix A. Using the transformation $s = x$, Eq. (17) can be transformed into

$$\frac{d^2\Phi_n(s)}{ds^2} - \left(\frac{-a^2 s + \nu(\nu+1)}{s^2} + \kappa_n^2\right)\Phi_n(s) = 0. \tag{19}$$

By comparing this equation with Eq. (A1) in the NU method of Appendix A, we get $\tilde{\tau}(s) = 0$, $\sigma(s) = s$, and $\tilde{\sigma}(s) = a^2 s - \nu(\nu+1) - \kappa_n^2 s^2$.

For further development of the theory, we introduce a function $\Pi(s)$ as [see Eq. (12) of Ref. [21]]

$$\Pi(s) = A(s) \pm \sqrt{A^2(s) - \tilde{\sigma}(s) + k\sigma(s)}\,, \tag{20}$$

where $A(s) = [\sigma'(s) - \tilde{\tau}(s)]/2$. Here, k is determined from the fact that the discriminant associated with this equation should be zero so that the expression inside the square root in this equation can be rearranged as the square of a polynomial. From Eq. (20), we have four possible values of $\Pi(s)$ as [10, 22]

$$\Pi(s) = \begin{cases} \kappa_n s + \nu, & \textbf{for} \quad k = k_1 \\ -(\kappa_n s + \nu + 1), & \textbf{for} \quad k = k_1 \\ \kappa_n s - \nu - 1, & \textbf{for} \quad k = k_2 \\ -\kappa_n s + \nu, & \textbf{for} \quad k = k_2\,, \end{cases} \tag{21}$$

where

$$k_1 = a^2 + 2\kappa_n(\nu + 1/2)\,, \qquad k_2 = a^2 - 2\kappa_n(\nu + 1/2)\,. \tag{22}$$

For the polynomial of $\tau(s) = \tilde{\tau}(s) + 2\Pi(s)$, $d\tau(s)/ds$ takes a negative value [23] and

http://dx.doi.org/10.5772/64007

$$\Pi(s) = -\kappa_n s + \nu, \tag{23}$$

with $k = k_2$. From the relation (see Appendix A)

$$\lambda = k + \Pi'(s), \tag{24}$$

λ can be expressed as

$$\lambda = a^2 - 2\kappa_n(\nu + 1). \tag{25}$$

For the case of $k_2 = a^2 - 2\kappa_n(\nu + 1/2)$, we have [23]

$$\lambda_n = 2n\kappa_n. \tag{26}$$

Now, let us equate Eq. (25) with Eq. (26) such that

$$2n\kappa_n = a^2 - 2\kappa_n(\nu + 1). \tag{27}$$

Then, by inserting the first and the second relations in Eq. (18) into the above equation, we easily confirm that the eigenvalues are given in the form

$$E_n = -\frac{\eta_0^2}{4\gamma_0\hbar^2}(n + \nu + 1)^{-2}, \tag{28}$$

where $n = 0, 1, 2, \cdots$. These are bound-state eigenvalues satisfying the boundary conditions [24]. This consequence agrees well with the report of Ref. [8] performed without using the NU method. To find eigenfunctions, we first need to determine the weight function $\rho(x)$ in Appendix A. Using Eq. (A5) in Appendix A and considering the condition in Eq. (23), we get $\rho(x) = \exp(-2\kappa_n x)x^{2\nu+1}$. Substituting this into Eq. (A6), we obtain the unnormalized values of z_n as [z_n is defined in Eq. (A2).]:

$$z_n(x) = L_n^{2\nu+1}(2\kappa_n x), \tag{29}$$

where $L_n^{2\nu+1}$ is the associated Laguerre polynomials [25] and C_n is the normalization factor. Now, using Eq. (A7) in Appendix A, we find

$$u_n(x) = \exp(-\kappa_n x)x^{\nu}. \tag{30}$$

Finally, regarding Eq. (A2) in Appendix A for bound states, the eigenfunctions of the invariant I_0, that are finite for all x, have the form

$$\Phi_{n\nu}(x) = C_n \exp(-\kappa_n x)x^{\nu} L_n^{2\nu+1}(2\kappa_n x). \tag{31}$$

where C_n is the normalization constant. By determining the exact formulae of C_n from the well-known condition

$$\int_0^{\infty} \Phi_{n\nu}(x)\Phi_{n\nu}(x)dx = 1, \tag{32}$$

the corresponding normalized wave functions are found to be

$$\begin{aligned}\Phi_{n\nu}(x) = &\left[\frac{n!}{2\Gamma(n+2\nu+1)!}\right]^{1/2} \frac{1}{(n+\nu+1)^{\nu+2}}\left[\frac{\eta_0}{\gamma_0\hbar^2}\right]^{\nu+3/2} \\ &\times x^{\nu}\exp\left[\frac{-\eta_0}{2\gamma_0\hbar^2(n+\nu+1)}x\right] L_n^{2\nu+1}\left(\frac{\eta_0}{\gamma_0\hbar^2(n+\nu+1)}x\right).\end{aligned} \tag{33}$$

Because the eigenstates of $I(x,p,t)$ are given by $\varphi_{n\nu}(x,t) = U^{-1}\Phi_{n\nu}(x)$, the normalized wave functions are evaluated as

$$\begin{aligned}\psi_{n\nu}(x,t) &= U^{-1}\Phi_{n\nu}(x)\exp[i\theta_{n\nu}(t)] \\ &= \left[\frac{n!\eta_0}{2\gamma_0\hbar^2(n+2\nu+1)!}\right]^{1/2} \frac{1}{(n+\nu+1)^{\nu+2}}\left[\frac{\eta(t)}{\gamma(t)\hbar^2}\right]^{\nu+1} \\ &\times x^{\nu}\exp\left[\frac{-i\beta(t)}{2\hbar\gamma(t)}x^2\right]\exp\left[\frac{-\eta(t)}{2\gamma(t)\hbar^2(n+\nu+1)}x\right] \\ &\times L_n^{2\nu+1}\left(\frac{\eta(t)}{\gamma(t)\hbar^2(n+\nu+1)}x\right)\exp[i\theta_{n\nu}(t)].\end{aligned} \tag{34}$$

There still remains the problem of finding the phases $\theta_{n\nu}(t)$ which satisfy Eq. (11). By carrying out the unitary transformation by means of $U(t)$, Eq. (11) becomes

http://dx.doi.org/10.5772/64007

$$\frac{d\theta_{nv}(t)}{dt} = -\frac{1}{2\hbar\mu(t)\gamma(t)}\langle\Phi_{nv}(x)\,|\,I_0(x,p)\,|\,\Phi_{nv}(x)\rangle. \tag{35}$$

Then, with the help of Eq. (28), this equation can be easily evaluated and, consequently, we obtain the phase factors in the form

$$e^{i\theta_{nv}(t)} = \exp\left[\frac{i\eta_0^2}{8\gamma_0\hbar^3(n+\nu+1)^2}\int_0^t\frac{1}{\mu(t')\gamma(t')}dt'\right]. \tag{36}$$

Now, by substituting Eq. (36) into Eq. (34), we find the exact nth-order solutions of the Schrödinger equation associated to the Hamiltonian $H(x,p,t)$. Eq. (34) is the full wave functions in the original system and agrees with the results of the report given in Ref. [8], which is performed for a little different system using another method. The wave functions are interpreted as probability amplitudes for finding the particle in the potential. These functions are defined everywhere and possess general properties for physical meaning such as continuousness and infinite differentiability. On the basis of the wave functions, various quantum properties of the system, such as expectation values of physical observables, energy eigen spectrum, and the uncertainty relation, can be investigated.

Let us see for a particular case that $\mu(t)$ is given by [10]

$$\mu(t) = m_0(1+\varepsilon t) \tag{37}$$

where m_0 and ε are positive real constants. In this case, Eq. (9) can be evaluated to be

$$F(t) = \frac{\beta_0\ln(1+\varepsilon t)}{m_0\varepsilon}\left(1-\frac{\beta_0}{2m_0\varepsilon\gamma_0}\ln(1+\varepsilon t)\right). \tag{38}$$

In this formula, we have used $\alpha_0 = \beta_0^2/\gamma_0$ according to the condition $\omega_0 = 0$ (see Section 2). If we substitute Eq. (38) in Eqs. (7) and (8), we have full expressions of $\gamma(t)$ and $\eta(t)$. By using $\gamma(t)$ obtained in such a way, the integration given in Eq. (36) can be fulfilled, and this results in

$$\int_0^t\frac{1}{\mu(t')\gamma(t')}dt' = \frac{\ln(1+\varepsilon t)}{m_0\varepsilon\gamma_0-\beta_0\ln(1+\varepsilon t)}. \tag{39}$$

Besides, Eq. (10) becomes

$$Z(t)=\frac{\eta_0}{2\gamma_0 m_0(1+\varepsilon t)}\left[1-2\frac{\beta_0\ln(1+\varepsilon t)}{m_0\varepsilon\gamma_0}\left(1-\frac{\beta_0}{2m_0\varepsilon\gamma_0}\ln(1+\varepsilon t)\right)\right]^{-1/2}. \tag{40}$$

If we choose $\beta_0 = m_0\gamma_0\varepsilon$ and $\eta_0 = 2m_0\gamma_0 Z_0$ where Z_0 is a real constant, Eq. (40) reduces to

$$Z(t)=\frac{Z_0}{(1+\varepsilon t)[1-\ln(1+\varepsilon t)]}, \tag{41}$$

within the time interval $0 \le t < (e-1)/\varepsilon$. Notice that Eq. (1) with Eqs. (37) and (41) is the same as Eq. (1) of Ref. [10]. Hence, we can confirm that the system treated in Ref. [10] is a particular case of a more general system that is studied in this chapter.

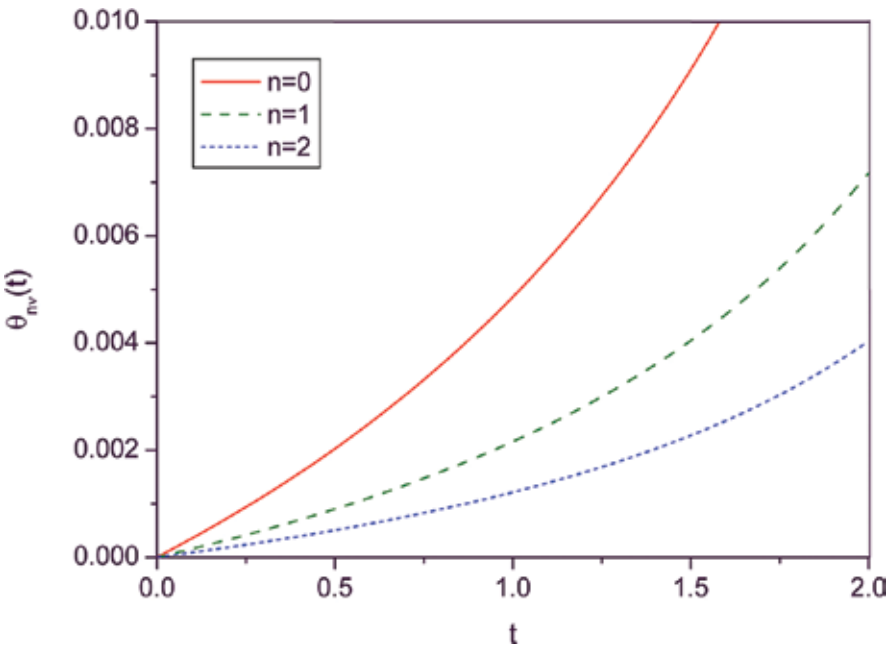

Figure 1. Time evolution of the phase $\theta_{nv}(t)$ for several different values of n. This is the case when $\mu(t)$ is given by Eq. (37). We have used $\beta_0 = 1, \gamma_0 = 3, \eta_0 = 1, m_0 = 1, \nu = 1, \hbar = 1$, and $\varepsilon = 0.1$.

Considering the relation given in Eq. (39), we have plotted the phase given in Eq. (36) in **Figures 1** and **2** as a function of time. From **Figure 1**, the increment of $\theta_{nv}(t)$ in time becomes

http://dx.doi.org/10.5772/64007

smaller as the quantum number n increases. We can confirm from **Figure 2** that the increment of $\theta_{nv}(t)$ also becomes smaller as ε increases.

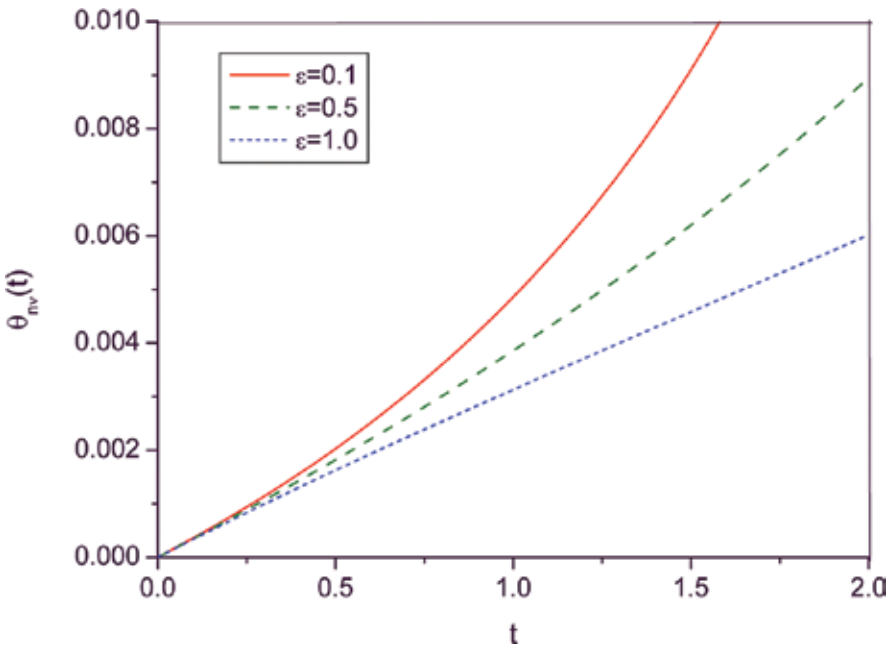

Figure 2. Time evolution of the phase $\theta_{nv}(t)$ for several different values of ε, where $\mu(t)$ is given by Eq. (37). We have used $\beta_0 = 1, \gamma_0 = 3, \eta_0 = 1, m_0 = 1, \nu = 1, \hbar = 1,$ and $n = 0$.

4. Conclusion

The invariant operator method and unitary transformation method were used in order to derive the quantum solutions of a time-dependent singular potential system that is described by the Hamiltonian given in Eq. (1). The quadratic invariant operator of the system has been determined from the use of its definition as shown in Eq. (3). The wave functions that satisfy the Schrödinger equation are given by multiplying the eigenstates $\varphi_n(t)$ of the invariant operator and the phase factors $e^{i\theta_n(t)}$ [see Eq. (34) with Eq. (12)]. By using the unitary operator, the original invariant operator $I(t)$ which is a time function was transformed to a simple form I_0 that is not a function of time. The NU method was used to derive the eigenstates of I_0. The eigenstates of I were derived from the inverse transformation of the eigenstates of I_0. The phases of the system were also derived from a fundamental relation in the framework of the

invariant operator theory. Through these procedures, the whole wave functions of the system as well as the eigenvalues of the invariant operator were obtained as shown in Eq. (34).

During the derivation of quantum solutions of the system, no approximation or perturbation methods were used. In fact, the merit of the invariant operator method for investigating quantization problem of TDHSs is that the corresponding quantum results are exact [3, 4]. Several methods for numerical treatment of time-dependent Schrödinger equations are known. If we enumerate some of them, they are the finite difference time domain (FDTD) method [26–31], the discretization method that takes advantage of the asymptotic behavior correspondence (ABC) [32, 33], and the discrete local discontinuous Galerkin method [34]. In particular, the FDTD method has been widely applied to obtain numerical solutions of mechanical problems of dynamical systems including Maxwell-Schrödinger equations for electromagnetic fields [30, 31]. If the methods for deriving numerical solutions of the Schrödinger equation for singular potential systems would be known in the future, it will be possible to compare our results developed in this chapter with them, leading to deepen the knowledge on quantum characteristics of relevant systems.

Appendices

Appendix A: Summary of the Nikiforov-Uvarov method

In this appendix, we introduce a useful method for solving Eq. (17) in the text, which is known as the NU method. This is useful for deriving the solutions of the Schrödinger-like second-order differential equations that play central roles in studying many important problems of theoretical physics. We first start from an appropriate coordinate transformation $s = s(x)$ for an arbitrary function g that satisfies the differential equation [19]:

$$g''(s) + \frac{\tilde{\tau}(s)}{\sigma(s)} g'(s) + \frac{\tilde{\sigma}(s)}{\sigma^2(s)} g(s) = 0\,, \tag{A1}$$

where $\sigma(s)$ and $\tilde{\sigma}(s)$ are some polynomials which at most are the second degree, and $\tilde{\tau}(s)$ is a polynomial of the first degree. A large part of special orthogonal polynomials [19] necessary in developing physics can be represented in the form of Eq. (A1). By expressing

$$g_n(s) = u_n(s) z_n(s)\,, \tag{A2}$$

where $u_n(s)$ are appropriate functions that will be chosen depending on the system. Eq. (A1) can be reduced into an equation of the following hypergeometric type [21]:

$$\sigma(s) z_n'' + \tau(s) z_n' + \lambda_n z_n = 0\,, \tag{A3}$$

where $\tau(s) = \tilde{\tau}(s) + 2\Pi(s)$ and λ_n are constants given in the form [23]

$$\lambda_n = -n\tau'(s) - n(n-1)\sigma''(s)/2. \tag{A4}$$

Notice that the derivative of $\tau(s)$ should be negative, while λ_n is obtained from a particular solution of the form $z(s) = z_n(s)$ which is a polynomial of degree n.

In terms of the weight function $\rho(s)$ that satisfies the condition [19]

$$\frac{d[\sigma(s)\rho(s)]}{ds} - \tau(s)\rho(s) = 0, \tag{A5}$$

the hypergeometric-type function $y_n(s)$ is given by [21]:

$$z_n(s) = C_n\rho^{-1}(s)\frac{d^n[\sigma^n(s)\rho(s)]}{ds^n}, \tag{A6}$$

where C_n is the normalization constant. This is known as the Rodrigues relation. Notice that Eq. (A5) is obtained from Eq. (20).

The relationship between λ and k introduced in the expression of Eq. (20) is $k = \lambda - \Pi'(s)$. Regarding this point, an appropriate formula for u_n can be evaluated from the condition [21]

$$u'(s) - \Omega(s)u(s) = 0, \tag{A7}$$

where $\Omega(s) = \Pi(s)/\sigma(s)$. For more details of the NU method, see Refs. [10, 19–23].

Author details

Salah Menouar[1] and Jeong Ryeol Choi[2*]

*Address all correspondence to: choiardor@hanmail.net

1 Laboratory of Optoelectronics and Compounds (LOC), Department of Physics, Faculty of Science, University of Ferhat Abbas Setif 1, Algeria

2 Department of Radiologic Technology, Daegu Health College, Yeongsong-ro, Buk-gu, Daegu, Republic of Korea

References

[1] Lewis, H. R. Jr. Classical and quantum systems with time-dependent harmonic-oscillator-type Hamiltonians. Phys. Rev. Lett., Vol. 18, No. 13, 510–512 (1967).

[2] Abdalla, M. S. Quantum treatment of the time-dependent coupled oscillators. J. Phys. A: Math. Gen., Vol. 29, No. 9, 1997–2012 (1996).

[3] Menouar, S., Maamache, M. & Choi, J. R. An alternative approach to exact wave functions for time-dependent coupled oscillator model of charged particle in variable magnetic field. Ann. Phys., Vol. 325, No. 8, 1708–1719 (2010).

[4] Menouar, S., Maamache, M. & Choi, J. R. The time-dependent coupled oscillator model for the motion of a charged particle in the presence of a time-varying magnetic field. Phys. Scr., Vol. 82, No. 6, 065004(1–7) (2010).

[5] Dodonov, V. V., Malkin, I. A. & Manko, V. I. Even and odd coherent states and excitations of a singular oscillator. Physica, Vol. 72, No. 3, 597–615 (1974).

[6] Choi, J. R. & Gweon, B. H. Operator method for a nonconservative harmonic oscillator with and without singular perturbation. Int. J. Mod. Phys. B, Vol. 16, No. 31, 4733–4742 (2002).

[7] Dodonov, V. V., Man'ko, V. I. & Rosa, L. Quantum singular oscillator as a model of a two-ion trap: an amplification of transition probabilities due to small-time variations of the binding potential. Phys. Rev. A, Vol. 57, No. 4, 2851–2858 (1998).

[8] Menouar, S., Maamache, M., Saadi, Y. & Choi, J. R. Exact wavefunctions for a time-dependent Coulomb potential. J. Phys. A: Math. Theor., Vol. 41, No. 21, 215303(1–11) (2008).

[9] Menouar, S., Maamache, M., Choi, J. R. & Sever, R. On the quantization of one-dimensional nonstationary Coulomb potential system. J. Phys. Soc. Japan, Vol. 81, No. 6, 064003(1–5) (2012).

[10] Menouar, S. & Choi, J. R. Quantization of time-dependent singular potential systems in one-dimension by using the Nikiforov-Uvarov method. J. Korean Phys. Soc., Vol. 67, No. 7, 1127–1132 (2015).

[11] Choi, J. R. & Oh, J. Y. Comparison of corrected wave functions associated to two different approaches for the time-dependent Hamiltonian systems involving (1/x)p + p(1/x) term. Int. J. Theor. Phys., Vol. 46, No. 10, 2591–2598 (2007).

[12] Choi, J. R. Exact wave functions of time-dependent Hamiltonian systems involving quadratic, inverse quadratic, and (1/x)p + p(1/x) terms. Int. J. Theor. Phys., Vol. 42, No. 4, 853–861 (2003).

http://dx.doi.org/10.5772/64007

[13] Menouar, S., Maamache, M., Bekkar, H. & Choi, J. R. Gaussian wave packet for time-dependent Hamiltonian systems involving quadratic, inverse quadratic, and (1/x)p + p(1/x) terms. J. Korean Phys. Soc., Vol. 58, No. 1, 154–157 (2011).

[14] Frank, W. M., Land, D. J. & Spector, R. M. Singular potentials. Rev. Mod. Phys., Vol. 43, No. 1, 36–96 (1971).

[15] Yoshida, A. Ultra-relativistic Hamiltonian with various singular potentials. arXiv:hep-th/9908145v1 (1999).

[16] Ichinose, T. On three magnetic relativistic Schrödinger operators and imaginary-time path integrals. Lett. Math. Phys., Vol. 101, No. 3, 323–339 (2012).

[17] Plesset, M. The Dirac electron in simple fields. Phys. Rev., Vol. 41, No. 3, 278–290 (1932).

[18] Lewis, H. R. Jr. & Riesenfeld, W. B. An exact quantum theory of the time-dependent harmonic oscillator and of a charged particle in a time-dependent electromagnetic field. J. Math. Phys., Vol. 10, No. 8, 1458–1473 (1969).

[19] Nikiforov, A. F. & Uvarov, V. B. Special Functions of Mathematical Physics. Birkhäuser Verlag Basel, Germany, 1988.

[20] Berkdemir, C. Application of the Nikiforov-Uvarov method in quantum mechanics, in Theoretical Concepts of Quantum Mechanics, M. R. Pahlavani (Ed.), InTech, Rijeka, 2012.

[21] Yacsuk, F., Berkdemir, C. & Berkdemir, A. Exact solutions of the Schrödinger equation with non-central potential by the Nikiforov-Uvarov method. J. Phys. A: Math. Gen., Vol. 38, No. 29, 6579–6586 (2005).

[22] Ikhdair, S. M. & Sever, R. Polynomial solution of PT-/non-PT-symmetric and non-Hermitian generalized Woods-Saxon potential via Nikiforov-Uvarov method. Int. J. Theor. Phys., Vol. 46, No. 6, 1643–1665 (2007).

[23] Ikhdair, S. M. & Sever, R. Polynomial solution of non-central potentials. arXiv:quant-ph/0702186v1 (2007).

[24] Liboff, R. L. Introductory Quantum Mechanics, 4th ed. Addison Wesley, San Fransisco, CA, 2003.

[25] Erdély, A. Higher Transcendental Functions, Vol. II, McGraw–Hill, New York, 1953.

[26] Bigaouette, N., Ackad, E. & Romunno, L. Nonlinear grid mapping applied to an FDTD-based, multi-center 3D Schrödinger equation solver. Comput. Phys. Commun., Vol. 183, No. 1, 38–45 (2012).

[27] Moxley III, F. I., Zhu, F. & Dai, W. A generalized FDTD method with absorbing boundary condition for solving a time-dependent linear Schrödinger equation. Am. J. Comput. Math., Vol. 2, No. 3, 163–172 (2012).

[28] Sudiarta, I. W. & Geldart, D. J. W. Solving the Schrödinger equation using the finite difference time domain method. J. Phys. A: Math. Theor., Vol. 40, No. 8, 1885–1896 (2007).

[29] Dai, W., Li, G., Nassar, R. & Su, S. On the stability of the FDTD method for solving a time-dependent Schrödinger equation. Numer. Methods Partial Differ. Eq., Vol. 21, No. 6, 1140–1154 (2005).

[30] Sui, W., Yang, J., Yun, X. H. & Wang, C. Including quantum effects in electromagnetic system—an FDTD solution to Maxwell-Schrödinger equations. Proceedings of the IEEE/MTT-S International Microwave Symposium (IEEE Xplore, New York), pp. 1979–1982 (2007).

[31] Ahmed, I. & Li, E. Simulation of plasmonics nanodevices with coupled Maxwell and Schrödinger equations using the FDTD method. Adv. Electromagn., Vol. 1, No. 1, 76–83 (2012).

[32] Gordon, A., Jirauschekm, C. & Kartner, F. X. Numerical solver of the time-dependent Schrödinger equation with Coulomb singularities. Phys. Rev. A, Vol. 73, No. 4, 042505(1–10) (2006).

[33] Antoine, X. & Besse, C. Unconditionally stable discretization schemes of non-reflecting boundary conditions for the one-dimensional Schrodinger equation. J. Comput. Phys., Vol. 188, No. 1, 157–175 (2003).

[34] Wei, L., Zhang, X., Kumar, S. & Yildirim, A. A numerical study based on an implicit fully discrete local discontinuous Galerkin method for the time-fractional coupled Schrödinger system. Comput. Math. Appl., Vol. 64, No. 8, 2603–2615 (2012).

Chapter 4

Smoothing Solution for Discrete-Time Nonlinear Stochastic Optimal Control Problem with Model-Reality Differences

Sie Long Kek , Kok Lay Teo and
Mohd Ismail Abd Aziz

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64564

Abstract

In this chapter, the performance of the integrated optimal control and parameter estimation (IOCPE) algorithm is improved using a modified fixed-interval smoothing scheme in order to solve the discrete-time nonlinear stochastic optimal control problem. In our approach, a linear model-based optimal control problem with adding the adjustable parameters into the model used is solved iteratively. The aim is to obtain the optimal solution of the original optimal control problem. In the presence of the random noise sequences in process plant and measurement channel, the state dynamics, which is estimated using Kalman filtering theory, is smoothed in a fixed interval. With such smoothed state estimate sequence that reduces the output residual, the feedback optimal control law is then designed. During the computation procedure, the optimal solution of the modified model-based optimal control problem can be updated at each iteration step. When convergence is achieved, the iterative solution approaches to the correct optimal solution of the original optimal control problem, in spite of model-reality differences. Moreover, the convergence of the resulting algorithm is also given. For illustration, optimal control of a continuous stirred-tank reactor problem is studied and the result obtained shows the efficiency of the approach proposed.

Keywords: fixed-interval smoothing, Kalman filtering theory, model-reality differences, adjustable parameters, iterative solution

1. Introduction

Optimal control approach provides the solution in solving dynamic real-world practical problems. Particularly, the linear problems, which are disturbed by the random noise sequence, have been well-defined with application of the optimal state estimate in designing the optimal feedback control law. In such situation, the optimal state estimator and the optimal controller are designed separately to optimize and control the dynamical systems. This is called the separation principle [1–4]. By virtue of this principle, the research works on stochastic optimal control and applications are growing widely, see for examples, linear systems [5, 6], fleet composition problem [7], optimal parameter selection problems [8], Markov jump process [9], power management [10], multiagent systems [11], portfolio selection model [12], 2-DOF vehicle model [13], sensorimotor system [14], and advertising model [15].

In fact, the exact solution of stochastic optimal control problems is impossible to be obtained, especially for the problems involving nonlinear system dynamics. To obtain an optimal solution of the discrete-time nonlinear stochastic optimal control problem, the integrated optimal control and parameter estimation (IOCPE) algorithm has been proposed to solve this kind of the problem iteratively [16–18]. In this algorithm, the linear quadratic Gaussian (LQG) model is applied to a model-based optimal control problem, where the state estimation procedure is done using the Kalman filtering theory. Based on this model, the adjusted parameters are added into the model so as system optimization and parameter estimation are integrated interactively. On this basis, the differences between the real plant and the model used are measured repeatedly in order to update the optimal solution of the model used. On the other hand, the output that is measured from the real plant is fed back into the model used for the state estimator design. When the convergence is achieved, the iterative solution approaches to the true optimal solution of the original optimal control problem despite model-reality differences. This optimal solution is the optimal filtering solution, which is obtained using the IOCPE algorithm. The efficiency of the IOCPE algorithm has been proven in Refs. [16–18].

However, the output trajectory of the model, which is obtained from the IOCPE algorithm, is less accurate in estimating the exact output measurement of the original optimal control problem. In this chapter, our aim is to improve the IOCPE algorithm using the fixed-interval smoothing approach, where the output residual shall be reduced within an appropriate tolerance to generate a better output trajectory. In our model, the state dynamics, which is disturbed by Gaussian noise sequences, is estimated by using the Kalman filtering theory, and then it is smoothed in a fixed-interval estimation. With such state estimation procedure, we modify the estimation procedure so that a smoothed state estimate is predicted backward in time and is used in designing the feedback optimal control law. It is noticed that the output residual of this smoothed state estimate is smaller than the output residual that is obtained by using the Kalman filtering theory, see [17]. The procedure of the solution method discussed in this chapter is almost the same as that was presented in the study of Kek et al. [17], but the accuracy of the optimal solution with the modified fixed-interval smoothing would be definitely increased.

http://dx.doi.org/10.5772/64564

The structure of the chapter is outlined as follows. In Section 2, the description of a general discrete-time nonlinear stochastic optimal control problem and its simplified model-based optimal control problem is made. In Section 3, an expanded optimal control model is introduced, where system optimization and parameter estimation are integrated mutually. The feedback control law, which is incorporated with the Kalman filtering theory and the fixed-interval smoothing, is designed. Then, the iterative algorithm based on principle of model-reality differences is derived so that discrete-time nonlinear stochastic optimal control problem could be solved. In Section 4, a convergence result for the algorithm proposed is provided. In Section 5, an example of optimal control of a continuous stirred-tank reactor problem is illustrated. Finally, some concluding remarks are made.

2. Problem description

Consider a general class of the dynamical system given below:

$$x(k+1) = f(x(k),u(k),k) + G\omega(k) \tag{1a}$$

$$y(k) = h(x(k),k) + \eta(k) \tag{1b}$$

where $u(k) \in \Re^m, k = 0,1,...,N-1, x(k) \in \Re^n, k = 0,1,...,N$, and $y(k) \in \Re^p, k = 0,1,...,N$ are the control sequence, the state sequence, and the output sequence, respectively. $\omega(k) \in \Re^q, k = 0,1,...,N-1$, which is the process noise sequence, and $\eta(k) \in \Re^p, k = 0,1,...,N$, which is the measurement noise sequence, are stationary Gaussian white noise sequences with zero mean, and their covariance matrices are given by $Q_\omega \in \Re^{q \times q}$ and $R_\eta \in \Re^{p \times p}$, respectively. Here, both of these covariance matrices are positive definite matrices. In addition, $f: \Re^n \times \Re^m \times \Re \to \Re^n$ represents the real plant and $h: \Re^n \times \Re \to \Re^p$ is the real output measurement, which both are assumed to be continuously differentiable with respect to their respective arguments, whereas $G \in \Re^{n \times q}$ is a process coefficient matrix.

The initial state is

$$x(0) = x_0$$

where $x_0 \in \Re^n$ is a random vector with mean and covariance given, respectively, by

$$E[x(0)] = \bar{x}_0 \text{ and } E[(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^{\mathrm{T}}] = M_0.$$

Here, $M_0 \in \Re^{n \times n}$ is a positive definite matrix and $E[\cdot]$ is the expectation operator. It is assumed that initial state, process noise, and measurement noise are statistically independent.

Therefore, our aim is to find an admissible control sequence $u(k) \in \Re^m, k = 0, 1, ..., N-1$ subject to the dynamical system given in Eq. (1) such that the scalar cost function

$$J_0(u) = E[\varphi(x(N), N) + \sum_{k=0}^{N-1} L(x(k), u(k), k)] \tag{2}$$

is minimized, where $\varphi: \Re^n \times \Re \to \Re$ is the terminal cost and $L: \Re^n \times \Re^m \times \Re \to \Re$ is the cost under summation. It is assumed that these functions are continuously differentiable with respect to their respective arguments.

This problem is regarded as the discrete-time nonlinear stochastic optimal control problem and is referred to as Problem (P).

Notice that, in general, the exact solution of Problem (P) is unable to be obtained and estimating the state of the real plant by applying the nonlinear filtering theory is computationally demanding. Due to these reasons, a smoothing model-based optimal control problem, which is referred to as Problem (M), is proposed by

$$\begin{aligned} \min_{u(k)} J_m(u) = & \tfrac{1}{2} \hat{x}_s(N)^{\mathrm{T}} S(N) \hat{x}_s(N) + \gamma(N) \\ & + \sum_{k=0}^{N-1} (\tfrac{1}{2} (\hat{x}_s(k)^{\mathrm{T}} Q \hat{x}_s(k) + u(k)^{\mathrm{T}} R u(k)) + \gamma(k)) \end{aligned} \tag{3}$$

subject to

$$\begin{aligned} \hat{x}_s(k) &= \hat{x}(k) + K_s(k)(\hat{x}_s(k+1) - \bar{x}(k+1)) \\ \hat{y}_s(k) &= C\hat{x}_s(k) \end{aligned}$$

with the following state estimation procedure

$$\bar{x}(k+1) = A\hat{x}(k) + Bu(k) + \alpha_1(k) \tag{4a}$$

$$\hat{x}(k) = \bar{x}(k) + K_f(k)(y(k) - \bar{y}(k)) \tag{4b}$$

$$\bar{y}(k) = C\bar{x}(k) + \alpha_2(k) \tag{4c}$$

http://dx.doi.org/10.5772/64564

where $\hat{x}_s(k) \in \Re^n, k = 0, 1, ..., N$ and $\hat{y}_s(k) \in \Re^p, k = 0, 1, ..., N$ are, respectively, the smoothed state sequence and the smoothed output sequence. The matrices involved are given as follow: A is an $n \times n$ state transition matrix, B is an $n \times n$ control coefficient matrix, C is a $p \times n$ output coefficient matrix, $S(N)$ and Q are $n \times n$ positive semidefinite matrices, and R is a $m \times m$ positive definite matrix. The extra parameters $\alpha_1(k), k = 0, 1, ..., N-1, \alpha_2(k), k = 0, 1, ..., N$, and $\gamma(k), k = 0, 1, ..., N$ are introduced as adjustable parameters.

The state estimation procedure, which is given in (4a), (4b), and (4c), is obviously from the Kalman filtering theory, where $\hat{x}(k) \in \Re^n, k = 0, 1, ..., N-1$ and $\bar{x}(k) \in \Re^n, k = 0, 1, ..., N$ are, respectively, the filtered state sequence and the predicted state sequence, whereas $\bar{y}(k) \in \Re^p, k = 0, 1, ..., N$ is the expected output sequence. The filter and smoother gains, which are $K_f(k) \in \Re^{n \times p}$ and $K_s(k) \in \Re^{n \times n}$, are, respectively, given by

$$K_f(k) = M_x(k)C^{\mathrm{T}}M_y(k)^{-1} \tag{5a}$$

$$K_s(k) = P(k)A^{\mathrm{T}}M_x(k+1)^{-1} \tag{5b}$$

whereas the state error covariance matrices are

$$P(k) = M_x(k) - M_x(k)C^{\mathrm{T}}M_y(k)^{-1}CM_x(k) \tag{6a}$$

$$M_x(k+1) = AP(k)A^{\mathrm{T}} + GQ_\omega G^{\mathrm{T}} \tag{6b}$$

$$P_s(k) = P(k) + K_s(k)(P_s(k+1) - M_x(k+1))K_s(k)^{\mathrm{T}} \tag{6c}$$

and the output error covariance matrix is

$$M_y(k) = CM_x(k)C^{\mathrm{T}} + R_\eta \tag{6d}$$

with the boundary conditions $M_x(0) = M_0$ and $P_s(N) = M_x(N)$ The filtered state error covariance $P(k) \in \Re^{n \times n}$, the predicted state error covariance $M_x(k) \in \Re^{n \times n}$, the smoothed state error covariance $P_s(k) \in \Re^{n \times n}$, and the output error covariance $M_y(k) \in \Re^{p \times p}$ are positive definite matrices.

Here, the cost function given in Eq. (3) is evaluated from the expectation of the quadratic forms [2], both for random and deterministic terms with trace matrix $tr(\ )$, which is simplified by

a. $E\left[x(N)^{\mathrm{T}}S(N)x(N)\right] = tr\left(S(N)M_x(N)\right) + \bar{x}(N)^{\mathrm{T}}S(N)\bar{x}(N)$

b. $E\left[x(k)^{\mathrm{T}}Qx(k)\right] = tr\left(QM_x(k)\right) + \bar{x}(k)^{\mathrm{T}}Q\bar{x}(k)$

c. $E\left[u(k)^{\mathrm{T}}Ru(k)\right] = u(k)^{\mathrm{T}}Ru(k)$

d. $E[\gamma(k)] = \gamma(k)$, $E\left[\alpha_1(k)\right] = \alpha_1(k)$, and $E\left[\alpha_2(k)\right] = \alpha_2(k)$.

Follow from this simplification, the trace matrix terms that are depend on the state error covariance matrix are ignored in the model used since they are constant values. In such a way, the cost function of the linear model-based optimal control model could be evaluated.

Notice that the separation principle [1–4] is applied to solving Problem (M), where the optimal feedback control law and the optimal state estimate are designed separately as discussed in [16–18]. Further from this, the accuracy of the optimal state estimate is increased by smoothing the state estimate in the fixed interval [2, 4]. Then, based on this smoothed state estimate, the smoothing optimal control law is designed. On the other hand, the output measured from the real plant is fed back into the model used, in turn, to improve the state estimation procedure and to update the solution of the model used. Moreover, only solving Problem (M) without adding the adjusted parameters into the model used would not approximate to the optimal solution of Problem (P). Hence, by taking the adjusted parameters into the model used and solving Problem (M) iteratively, the correct optimal solution of the original optimal control problem could be obtained, in spite of model-reality differences.

3. Modified smoothing with model-reality differences

Now, let us introduce an expanded optimal control problem with smoothing state estimate, which is referred to as Problem (E), given below:

$$\begin{aligned}\min_{u(k)} J_e(u) = {} & \tfrac{1}{2}\hat{x}_s(N)^{\mathrm{T}}S(N)\hat{x}_s(N) + \gamma(N) \\ & + \sum_{k=0}^{N-1}\left(\tfrac{1}{2}(\hat{x}_s(k)^{\mathrm{T}}Q\hat{x}_s(k) + u(k)^{\mathrm{T}}Ru(k)\right) + \gamma(k)) \\ & + \tfrac{1}{2}r_1 \| v(k) - u(k) \|^2 + \tfrac{1}{2}r_2 \| z(k) - \hat{x}_s(k) \|^2\end{aligned} \tag{7}$$

subject to

$$\hat{x}_s(k) = \hat{x}(k) + K_s(k)(\hat{x}_s(k+1) - \bar{x}(k+1))$$

$$\hat{y}_s(k) = C\hat{x}_s(k)$$

http://dx.doi.org/10.5772/64564

$$\frac{1}{2}z(N)^{\mathrm{T}}S(N)z(N)+\gamma(N)=\varphi(z(N),N)$$

$$\frac{1}{2}(z(k)^{\mathrm{T}}Qz(k)+v(k)^{\mathrm{T}}Rv(k))+\gamma(k)=L(z(k),v(k),k)$$

$$Az(k)+Bv(k)+\alpha_1(k)=f(z(k),v(k),k)$$

$$Cz(k)+\alpha_2(k)=h(z(k),k)$$

$$v(k)=u(k)$$

$$z(k)=\hat{x}_s(k)$$

where $v(k)\in\Re^m, k=0,1,...,N-1$ and $z(k)\in\Re^n, k=0,1,...,N$ are introduced to separate the control and the smoothed state from the respective signals in the parameter estimation problem and $\|\cdot\|$ denotes the usual Euclidean norm. The terms $\frac{1}{2}r_1\|u(k)-v(k)\|^2$ and $\frac{1}{2}r_2\|\hat{x}_s(k)-z(k)\|^2$ are introduced such that the convexity is improved and the convergence of the iterative algorithm is enhanced. The main purpose of designing the algorithm in this way is to ensure that satisfying of the constraints $v(k)=u(k)$ and $z(k)=\hat{x}_s(k)$ is fulfilled at the end of the iterations. More specifically, applying the state estimate $z(k)$ and the control $v(k)$ for the computation in the parameter estimation and the matching schemes will increase the practical usage of the algorithm. Moreover, implementing the relevant smoothed state $\hat{x}_s(k)$ and control $u(k)$ that will be reserved for optimizing the model-based optimal control problem leads the iterative solution toward to the true optimal solution of the original optimal control problem.

Figure 1 shows the block diagram of the approach proposed. The methodology of the approach proposed is further discussed in the following sections.

From the block diagram in **Figure 1**, the definition of the principle of model-reality differences could be given.

Definition 3.1: Principle of model-reality differences is a unified framework, which integrates system optimization and parameter estimation interactively to define an expanded optimal

control problem, aims to give the correct optimal solution of the original optimal control problem by solving the model-based optimal control problem iteratively.

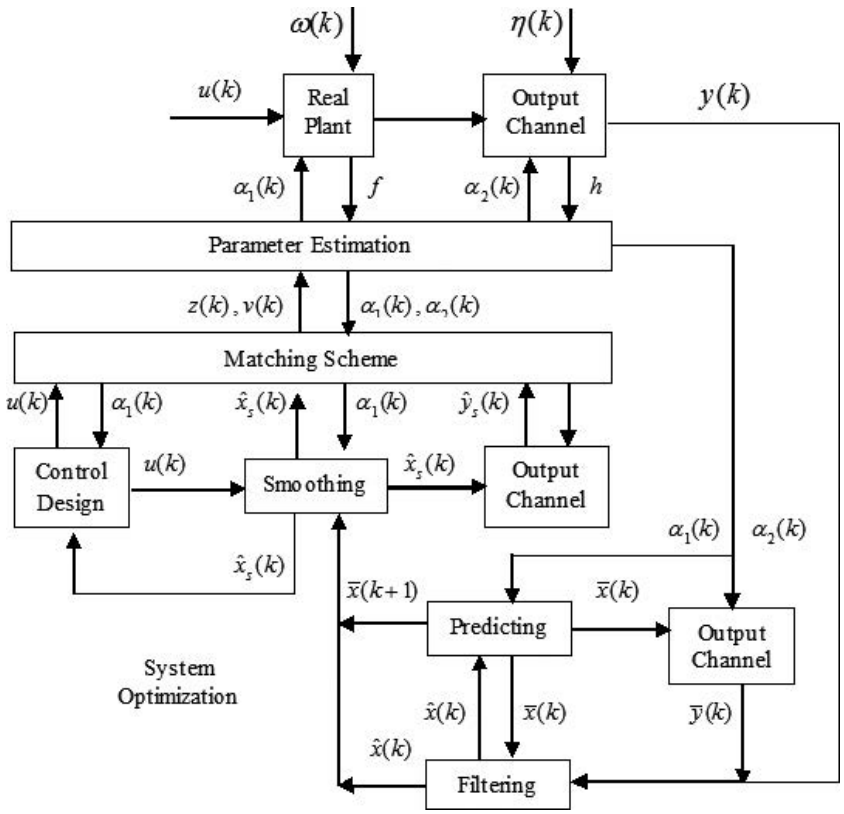

Figure 1. Block diagram of the approach proposed.

3.1. Optimality conditions

Define the Hamiltonian function for Problem (E) as follows:

$$\begin{aligned} H_e(k) &= \tfrac{1}{2}(\hat{x}_s(k)^{\mathrm{T}} Q \hat{x}_s(k) + u(k)^{\mathrm{T}} R u(k)) + \gamma(k) \\ &\quad + \tfrac{1}{2} r_1 \| v(k) - u(k) \|^2 + \tfrac{1}{2} r_2 \| z(k) - \hat{x}_s(k) \|^2 \\ &\quad - \lambda(k)^{\mathrm{T}} u(k) - \beta(k)^{\mathrm{T}} \hat{x}_s(k) \\ &\quad + q(k)^{\mathrm{T}} (C \hat{x}_s(k) - \hat{y}_s(k)) \\ &\quad + p(k+1)^{\mathrm{T}} (\hat{x}_s(k) - \hat{x}(k) - K_s(k)(\hat{x}_s(k+1) - \bar{x}(k+1))). \end{aligned} \tag{8}$$

Then, the augmented cost function becomes

http://dx.doi.org/10.5772/64564

$$\begin{aligned} J'_e(k) = & \tfrac{1}{2}\hat{x}_s(N)^{\mathrm{T}} S(N)\hat{x}_s(N) + \gamma(N) + \Gamma^{\mathrm{T}}(\hat{x}_s(N) - z(N)) \\ & + \xi(N)(\varphi(z(N),N) - \tfrac{1}{2}z(N)^{\mathrm{T}} S(N)z(N) - \gamma(N)) \\ & + \sum_{k=0}^{N-1} H_e(k) + \lambda(k)^{\mathrm{T}} v(k) + \beta(k)^{\mathrm{T}} z(k) \\ & + \xi(k)(L(z(k),v(k),k) - \tfrac{1}{2}(z(k)^{\mathrm{T}} Qz(k) + v(k)^{\mathrm{T}} Rv(k)) - \gamma(k)) \\ & + \mu(k)^{\mathrm{T}}(f(z(k),v(k),k) - Az(k) - Bv(k) - \alpha_1(k)) \\ & + \pi(k)^{\mathrm{T}}(h(z(k),k) - Cz(k) - \alpha_2(k)) \end{aligned} \tag{9}$$

where $p(k), q(k), \mu(k), \xi(k), \pi(k), \Gamma, \beta(k),$ and $\lambda(k)$ are the proper multipliers to be judged the value later.

The following necessary conditions for optimality are resulted when applying the calculus of variation [2, 4, 17] to the augmented cost function given in Eq. (9):

(a) Stationary condition:

$$Ru(k) + B^{\mathrm{T}} K_s(k) p(k+1) - \lambda(k) - r_1(v(k) - u(k)) = 0. \tag{10a}$$

(b) Smoothed costate equation:

$$p(k) = Q\hat{x}_s(k) + p(k+1) - \beta(k) - r_2(z(k) - \hat{x}_s(k)). \tag{10b}$$

(c) Smoothed state equation:

$$\hat{x}_s(k) = \hat{x}(k) + K_s(k)(\hat{x}_s(k+1) - \bar{x}(k+1)) \tag{10c}$$

with the boundary conditions $\hat{x}_s(N) = \bar{x}(N)$ and $p(N) = \Gamma$.

(d) Adjustable parameter equations:

$$\varphi(z(N),N) = \tfrac{1}{2}z(N)^{\mathrm{T}} S(N)z(N) + \gamma(N) \tag{11a}$$

$$L(z(k),v(k),k) = \tfrac{1}{2}(z(k)^{\mathrm{T}} Qz(k) + v(k)^{\mathrm{T}} Rv(k)) + \gamma(k) \tag{11b}$$

$$f(z(k),v(k),k) = Az(k) + Bv(k) + \alpha_1(k) \tag{11c}$$

$$h(z(k),k) = Cz(k) + \alpha_2(k). \tag{11d}$$

(e) Multiplier equations:

$$\Gamma - \nabla_{z(k)}\varphi + S(N)z(N) = 0 \tag{12a}$$

$$\lambda(k) + (\nabla_{v(k)}L - Rv(k)) + \left(\frac{\partial f}{\partial v(k)} - B\right)^{\mathrm{T}} \hat{p}(k+1) = 0 \tag{12b}$$

$$\beta(k) + (\nabla_{z(k)}L - Qz(k)) + \left(\frac{\partial f}{\partial z(k)} - A\right)^{\mathrm{T}} \hat{p}(k+1) = 0 \tag{12c}$$

with $\xi(k) = 1, \mu(k) = \hat{p}(k+1)$ and $\pi(k) = q(k) = 0$.

(f) Separable variables:

$$v(k) = u(k), z(k) = \hat{x}_s(k), \hat{p}(k) = p(k). \tag{13}$$

In view of these necessary optimality conditions, the conditions (10a), (10b), and (10c) define the modified model-based optimal control problem, the conditions (11a), (11b), (11c), and (11d) define the parameter estimation problem and the conditions (12a), (12b), and (12c) are used to compute the multipliers. They are further discussed as follows.

3.2. Modified model-based optimal control problem

The modified model-based optimal control problem, which is referred to as Problem (MM), is given below:

$$\begin{aligned}\min_{u(k)} J_{mm}(u) = {} & \tfrac{1}{2}\hat{x}_s(N)^{\mathrm{T}} S(N)\hat{x}_s(N) + \gamma(N) + \Gamma^{\mathrm{T}}\hat{x}_s(N) \\ & + \sum_{k=0}^{N-1} \tfrac{1}{2}(\hat{x}_s(k)^{\mathrm{T}} Q\hat{x}_s(k) + u(k)^{\mathrm{T}} Ru(k)) + \gamma(k) \\ & + \tfrac{1}{2}r_1 \| v(k) - u(k) \|^2 + \tfrac{1}{2}r_2 \| z(k) - \hat{x}_s(k) \|^2 \\ & - \lambda(k)^{\mathrm{T}} u(k) - \beta(k)^{\mathrm{T}} \hat{x}_s(k)\end{aligned} \tag{14}$$

subject to

http://dx.doi.org/10.5772/64564

$$\hat{x}_s(k) = \hat{x}(k) + K_s(k)(\hat{x}_s(k+1) - \bar{x}(k+1))$$

$$\hat{y}_s(k) = C\hat{x}_s(k).$$

From the outcome of Problem (E) and Problem (MM), the theorem of the smoothed optimal control law which is applied to solve Problem (MM) is described.

Theorem 3.1: Suppose the expanded optimal control law for Problem (E) exists. Then, this control law is the smoothed feedback control law for Problem (MM) given by

$$u(k) = -K(k)\hat{x}_s(k) + u_{ff}(k) \tag{15}$$

where

$$\begin{aligned} u_{ff}(k) = -(R_a + B^{\mathrm{T}}K_s(k)S(k+1)B)^{-1}(B^{\mathrm{T}}K_s(k)s(k+1) - \lambda_a(k) \\ + B^{\mathrm{T}}K_s(k)S(k+1)((A - K_s(k)^{-1})\hat{x}(k) + \alpha_1(k))) \end{aligned} \tag{16a}$$

$$K(k) = (R_a + B^{\mathrm{T}}K_s(k)S(k+1)B)^{-1}B^{\mathrm{T}}K_s(k)S(k+1)K_s(k)^{-1} \tag{16b}$$

$$S(k) = Q_a + S(k+1)(K_s(k)^{-1} - BK(k)) \tag{16c}$$

$$s(k) = S(k+1)((A - K_s(k)^{-1})\hat{x}(k) + Bu_{ff}(k) + \alpha_1(k)) + s(k+1) - \beta_a(k) \tag{16d}$$

with the boundary conditions $S(N)$ given and $s(N) = 0$, and

$$R_a = R + r_1 I_m;\ Q_a = Q + r_2 I_n;$$

$$\lambda_a(k) = \lambda(k) + r_1 v(k);\ \beta_a(k) = \beta(k) + r_2 z(k).$$

Proof: From the necessary optimality condition (10a), we have

$$R_a u(k) = -B^{\mathrm{T}}K_s(k)p(k+1) + \lambda_a(k). \tag{17}$$

Applying sweep method [2, 4],

$$p(k) = S(k)\hat{x}_s(k) + s(k) \tag{18}$$

we substitute Eq. (18) for $k = k + 1$ into Eq. (17), which yields

$$R_a u(k) = -B^{\mathrm{T}} K_s(k) S(k+1) x_s(k+1) - B^{\mathrm{T}} K_s(k) s(k+1) + \lambda_a(k). \tag{19}$$

Rewrite the smoothed state equation from Eq. (10c),

$$\hat{x}_s(k+1) = \bar{x}(k+1) + (K_s(k))^{-1}(\hat{x}_s(k) - \hat{x}(k)). \tag{20}$$

Then, substitute Eq. (20) into Eq. (19). After some algebraic manipulations, the smoothed control law (15) is obtained, where Eqs. (16a) and (16b) are satisfied.

From the smoothed costate equation (10b), we substitute Eq. (18) for $k = k + 1$ to give

$$p(k) = Q_a \hat{x}_s(k) + S(k+1)\hat{x}_s(k+1) + s(k+1) - \beta_a(k) \tag{21}$$

Consider Eq. (20) in Eq. (21), we obtain

$$p(k) = Q_a \hat{x}_s(k) + S(k+1)(\bar{x}(k+1) + (K_s(k))^{-1}(\hat{x}_s(k) - \hat{x}(k)) + s(k+1) - \beta_a(k). \tag{22}$$

By doing some algebraic manipulations, it is found that Eqs. (16c) and (16d) are satisfied after comparing to Eq. (18). This completes the proof.

From Eqs. (4a), (10c), and (15), the smoothed state equation becomes

$$\begin{aligned} \hat{x}_s(k) = (I_n - K_s(k)BK(k))^{-1}((I_n - K_s(k)A)\hat{x}(k) \\ + K_s(k)(\hat{x}_s(k+1) - Bu_{ff}(k) - \alpha_1(k))) \end{aligned} \tag{23}$$

and the smoothed output is measured from

$$\hat{y}_s(k) = C\hat{x}_s(k) \tag{24}$$

with the boundary condition $\hat{x}_s(N) = \bar{x}(k)$.

http://dx.doi.org/10.5772/64564

3.3. Parameter estimation

After solving Problem (MM), the defined separable variables given in Eq. (13) are used for the further computations. Particularly, in the parameter estimation problem, the differences between the real plant and the model used are taken into account in which the matching schemes are established. In view of this, the adjusted parameters, which are resulted from parameter estimation problem defined by Eq. (11), are calculated from

$$\alpha_1(k) = f(z(k), v(k), k) - Az(k) - Bv(k) \tag{25a}$$

$$\alpha_2(k) = h(z(k), k) - Cz(k) \tag{25b}$$

$$\gamma(N) = \varphi(z(N), N) - \tfrac{1}{2} z(N)^{\mathrm{T}} S(N) z(N) \tag{25c}$$

$$\gamma(k) = L(z(k), v(k), k) - \tfrac{1}{2}(z(k)^{\mathrm{T}} Q z(k) + v(k)^{\mathrm{T}} R v(k)) \tag{25d}$$

3.4. Computation of multipliers

The multipliers, which are related to the Jacobian matrix of the functions f and L with respect to $v(k)$ and $z(k)$, are computed from

$$\Gamma = \nabla_{z(k)} \varphi - S(N) z(N) \tag{26a}$$

$$\lambda(k) = -(\nabla_{v(k)} L - Rv(k)) - \left(\frac{\partial f}{\partial v(k)} - B \right)^{\mathrm{T}} \hat{p}(k+1) \tag{26b}$$

$$\beta(k) = -(\nabla_{z(k)} L - Qz(k)) - \left(\frac{\partial f}{\partial z(k)} - A \right)^{\mathrm{T}} \hat{p}(k+1) \tag{26c}$$

3.5. Iterative algorithm

From the previous sections, the derivation of equations and the formulation of the resulting algorithm are clearly discussed. Following from these discussions, a summary on this iterative algorithm is delivered as follows:

Data $Q, R, S(N), A, B, C, G, Q_{\omega}, R_{\eta}, M_0, \bar{x}_0, N, r_1, r_2, k_v, k_z, k_p, f, L, h, \varphi$. Note that A and B may be chosen through the linearization of f, and C is obtained from the linearization of h.

Step 0: Compute a nominal solution. Assume $\alpha_1(k) = 0, k = 0, 1, ..., N-1, \alpha_2(k) = 0, k = 0, 1, ..., N$, and $r_1 = r_2 = 0$. Calculate $K_f(k)$ and $K_s(k)$ from Eqs. (5a) and (5b), $P(k), M_x(k), P_s(k)$ and $M_y(k)$ from Eqs. (6a), (6b), (6c), and (6d) for the state estimation, and solve Problem (M) defined by Eq. (3) to obtain $u(k)^0, k = 0, 1, ..., N-1$, and $\hat{x}_s(k)^0, \hat{y}_s(k)^0, p(k)^0, k = 0, 1, ..., N$. Then, with $\alpha_1(k) = 0, k = 0, 1, ..., N-1, \alpha_2(k) = 0,\ k = 0, 1, ..., N$, and r_1, r_2 from data, calculate $K(k)$ and $S(k)$, respectively, from Eqs. (16b) and (16c). Set $i = 0, z(k)^0 = \hat{x}_s(k)^0, v(k)^0 = u(k)^0$ and $\hat{p}(k)^0 = p(k)^0$.

Step 1: Calculate the adjustable parameters $\alpha_1(k)^i,\ k = 0\ , 1, ...,\ N-1, \alpha_2(k)^i, k = 0, 1, ..., N, \gamma(k)^i, k = 0, 1, ..., N$, from Eq. (25). This is called the *parameter estimation* step.

Step 2: Compute the modifiers $\Gamma^i, \lambda(k)^i$ and $\beta(k)^i, k = 0, 1, ..., N-1$, from Eq. (26). This requires the partial derivatives of f, h and L with respect to $v(k)^i$ and $z(k)^i$.

Step 3: With the determined $\alpha_1(k)^i, \alpha_2(k)^i, \gamma(k)^i, \Gamma^i, \lambda(k)^i, \beta(k)^i, v(k)^i$, and $z(k)^i$, solve Problem (MM) defined by Eq. (14) using the result in Theorem 3.1. This is called the *system optimization* step.

a. Obtain $s(k)^i, k = 0, 1, ..., N$ by solving Eq. (16d) backward, and obtain $u_{ff}(k)^i, k = 0, 1, ..., N-1$ by solving Eq. (16a), either backward or forward.

b. Calculate the new control $u(k)^i, k = 0, 1, ..., N-1$ using Eq. (15).

c. Calculate the new state $\hat{x}_s(k)^i, k = 0, 1, ..., N$, using Eq. (23).

d. Calculate the new costate $p(k)^i, k = 0, 1, ..., N$, using Eq. (18).

e. Calculate the new output $\hat{y}_s(k)^i, k = 0, 1, ..., N$, using Eq. (24).

Step 4: Update the optimal smoothing solution of Problem (P) and test the convergence of the algorithm. For regulating convergence, a mechanism, which is a simple relaxation method, shall be provided and given by:

$$z(k)^{i+1} = z(k)^i + k_z(\hat{x}_s(k)^i - z(k)^i) \tag{27a}$$

$$v(k)^{i+1} = v(k)^i + k_v(u(k)^i - v(k)^i) \tag{27b}$$

$$\hat{p}(k)^{i+1} = \hat{p}(k)^i + k_p(p(k)^i - \hat{p}(k)^i) \tag{27c}$$

http://dx.doi.org/10.5772/64564

where k_v, k_z, k_p, range in the interval of $(0, 1]$, are scalar gains. If $z(k)^{i+1} = z(k)^i, k = 0, 1, ..., N$, and $v(k)^{i+1} = v(k)^i, k = 0, 1, ..., N-1$, within a given tolerance, stop; else repeat from Step 1 by setting $i = i + 1$.

Remarks:

a. The off-line computation, which is mentioned in Step 0, is done for the state estimator design, where $K_f(k), K_s(k), k = 0, 1, ..., N-1, M_x(k), M_y(k), k = 0, 1, ..., N, P(k), P_s(k), k = 0, 1, ..., N-1$ are computed, and for the control law design, where $K(k), k = 0, 1, ..., N-1, S(k), k = 0, 1, ..., N$ are calculated. In fact, these parameters are used for solving Problem (M) in Step 0 and for solving Problem (MM) in Step 3, respectively.

b. The variables $\gamma(k)^i, \alpha_1(k)^i, \alpha_2(k)^i, \Gamma^i, \lambda(k)^i, \beta(k)^i$, and $s(k)^i$ are initially zero in Step 0. Their computed values, where $\gamma(k)^i, \alpha_1(k)^i, \alpha_2(k)^i$ in Step 1, $\Gamma^i, \lambda(k)^i, \beta(k)^i$ in Step 2, and $s(k)^i$ in Step 3, would be changed from iteration to iteration.

c. The driving input $u_{ff}(k)$ in Eq. (16a) corrects the differences between the real plant and the model used, and it also drives the controller given in Eq. (15).

d. The state estimation without the control is done forward using the Kalman filtering, and then it is followed by the fixed-interval smoothing backward in order to design the feedback control law.

e. Problem (P) is not necessary to have a cost function in quadratic criterion or to be a linear problem.

f. The equations $z(k)^{i+1} = z(k)^i$ and $v(k)^{i+1} = v(k)^i$ can be definitely required to satisfy for the converged state estimate sequence and the converged optimal control sequence. On this point of view, the following averaged 2-norms are computed and, then, they are compared with a given tolerance to verify the convergence of $v(k)$ and $z(k)$:

$$\| v^{i+1} - v^i \|_2 = \left(\frac{1}{N-1} \sum_{k=0}^{N-1} \| v(k)^{i+1} - v(k)^i \| \right)^{1/2} \tag{28a}$$

$$\| z^{i+1} - z^i \|_2 = \left(\frac{1}{N} \sum_{k=0}^{N} \| z(k)^{i+1} - z(k)^i \| \right)^{1/2} \tag{28b}$$

g. The relaxation scalars (k_v, k_z, k_p) are the step-sizes in regulating the convergence mechanism. These scalars could be normally chosen as a certain value in the range of (0, 1], but this choice may not provide the optimal number of iterations. Hence, it is important to note that the optimal choice of these scalars $k_v, k_z, k_p \in (0, 1]$ would be problem dependent.

As a rule of this case, the algorithm (from Step 1 to Step 4) is required to run few times. Initially, for first run of the algorithm (from Step 1 to Step 4), these scalars are set at $k_v = k_z = k_p = 1$, and then, with different values chosen from 0.1 to 0.9, the algorithm is run again. The value with the optimal number of iterations can be determined after that. Applying the parameters r_1 and r_2 is to enhance the convexity such that the convergence of the algorithm can be improved.

4. Convergence analysis

In this section, the convergence of the algorithm is discussed. The following assumptions are needed:

The derivatives of f, L and h exist.

The solution (u^*, x^*, y^*) is the optimal solution to Problem (P). That is, the optimal smoothing solution.

The convergence result is presented in Theorem 4.1, while the accuracy of the smoothed state in term of state error covariance is proven in Corollary 4.1.

Theorem 4.1: The converged solution of Problem (M) is the correct optimal smoothing solution of Problem (P).

Proof: Consider the real plant and the output measurement of Problem (P) with the exact optimal smoothing solution (u^*, x^*, y^*) as given below:

$$x^*(k+1) = f(x^*(k), u^*(k), k) \text{ and } y^*(k) = h(x^*(k), k) \tag{29}$$

In Problem (M), the model used consists of

$$\hat{x}^c(k) = \bar{x}^c(k) + K_f(k)(y(k) - \bar{y}^c(k)) \tag{30a}$$

$$\bar{x}^c(k+1) = A\hat{x}^c(k) + Bu^c(k) + \alpha_1(k) \tag{30b}$$

$$\bar{y}^c(k) = C\bar{x}^c(k) + \alpha_2(k) \tag{30c}$$

$$\hat{x}_s^c(k) = \hat{x}^c(k) + K_s(k)(\hat{x}_s^c(k+1) - \bar{x}^c(k+1)) \tag{30d}$$

http://dx.doi.org/10.5772/64564

$$\hat{y}_s^c(k) = C\hat{x}_s^c(k) \tag{30e}$$

where $u^c(k), \hat{x}_s^c(k), \hat{x}^c(k), \bar{x}^c(k), \hat{y}_s^c(k)$, and $\bar{y}^c(k)$ are, respectively, the converged sequences for control law, smoothed state estimate, filtered state estimate, expected state estimate, smoothed output, and expected output. Here, $y(k)$ is the output measured from the real plant.

Applying the adjusted parameters $\alpha_1(k)$ and $\alpha_2(k)$, which are given by

$$\alpha_1(k) = f(z(k), v(k), k) - Az(k) - Bv(k) \text{and}$$

$$\alpha_2(k) = h(z(k), k) - Cz(k),$$

into the model used given by Eq. (30b) and (30c), the differences between the real plant and the model used can be measured at each iteration. Moreover, at the end of iteration, from Eqs. (29) and (30a) – (30e) yields

$$\hat{x}_s^c(k+1) = f(z(k), v(k), k) \text{and} \, \hat{y}_s^c(k) = h(z(k), k)$$

which $v(k) = u^c(k)$ and $z(k) = \hat{x}_s^c(k) = \hat{x}^c(k)$ are satisfied. Hence, this implies that

$$u^c(k) = u^*(k),\ \hat{x}_s^c(k) = x^*(k),\ \hat{y}_s^c(k) = y^*(k)$$

This completes the proof.

Corollary 4.1: The smoothed state error covariance is the smallest among the values of state error covariance.

Proof: From Eq. (6), it is clear that the filtered state error covariance $P(k)$ is less than the predicted state error covariance $M_x(k)$. That is, $P(k) < M_x(k)$. Now, to prove $P_s(k) < P(k)$,, we shall show that $P_s(k+1) < M_x(k+1)$. Consider the boundary condition $P_s(N) = M_x(N)$ and taking $k = N - 1$, we have

$$P_s(N-1) = P(N-1) < M_x(N-1).$$

For $k = N - 2$, it shows that

$$P_s(N-2) < P(N-2) < M_x(N-2).$$

This statement can be deduced that

$$P_s(k+1) - M_x(k+1) < 0 \text{ for } k = k+1.$$

Thus, we conclude that

$$P_s(k) < P(k) < M_x(k), \quad k = 0,1,...,N-2,$$

which shows the accuracy of the smoothed state estimate. This completes the proof.

5. Illustrative example

Consider a continuous stirred-tank reactor problem [19], which consists of the state difference equations

$$x_1(k+1) = x_1(k) - 0.02(x_1(k)+0.25) + 0.01(x_2(k)+0.5)\exp\left[\frac{25x_1(k)}{x_1(k)+2}\right] - 0.01(x_1(k)+0.25)u(k) + \omega_1(k)$$

$$x_2(k+1) = 0.99x_2(k) - 0.005 - 0.01(x_2(k)+0.5)\exp\left[\frac{25x_1(k)}{x_1(k)+2}\right] + \omega_2(k)$$

for $k = 0, ..., 77$, and the output measurement $y(k) = x_1(k) + \eta(k)$. The initial state $x(0) = x_0$ is a random vector with mean and covariance given, respectively, by $\bar{x}_1(0) = 0.05$, $\bar{x}_2(0) = 0$, and $M_0 = 10^{-2}I_2$.

Here, $\omega(k) = [\omega_1(k) \ \ \omega_2(k)]^{\mathrm{T}}$ and $\eta(k)$ are Gaussian white noise sequences with their respective covariance given by $Q_\omega = 10^{-3}I_2$ and $R_\eta = 10^{-3}$. The expected cost function

$$J_0(u) = 0.5\sum_{k=0}^{N-1} E[(x_1(k))^2 + (x_2(k))^2 + 0.1(u(k))^2]$$

is to be minimized over the state difference equations and the output measurement.

This problem is referred to as Problem (P).

http://dx.doi.org/10.5772/64564

To obtain the optimal smoothing solution of Problem (P), we simplify the plant dynamics of Problem (P) and refer it as Problem (M), given by

$$\min_{u(k)} J_m(u) = \frac{1}{2}\sum_{k=0}^{N-1}[(\hat{x}_s(k))^2 + 0.1(u(k))^2 + 2\gamma(k)]$$

subject to

$$\hat{x}_s(k) = \hat{x}(k) + K_s(k)(\hat{x}_s(k+1) - \bar{x}(k+1))$$

$$\hat{y}_s(k) = C\hat{x}_s(k)$$

with

$$\hat{x}(k) = \bar{x}(k) + K_f(k)(y(k) - \bar{y}(k))$$

$$\begin{bmatrix} \bar{x}_1(k+1) \\ \bar{x}_2(k+1) \end{bmatrix} = \begin{bmatrix} 1.0895 & 0.0184 \\ -0.1095 & 0.9716 \end{bmatrix}\begin{bmatrix} \hat{x}_1(k) \\ \hat{x}_2(k) \end{bmatrix} + \begin{bmatrix} -0.003 \\ 0.000 \end{bmatrix}u(k) + \begin{bmatrix} \alpha_{11}(k) \\ \alpha_{12}(k) \end{bmatrix}$$

$$\bar{y}(k) = \bar{x}_1(k) + \alpha_2(k)$$

with the initial condition $\bar{x}(0) = \bar{x}_0$ and the boundary value $\hat{x}_s(N) = \bar{x}(N)$. Here, $\gamma(k), \alpha_2(k)$ and $\alpha_1(k) = [\alpha_{11}(k)\ \ \alpha_{12}(k)]^T$ are the adjusted parameters.

Model	Iteration number	Elapsed time	Initial cost	Final cost	Output residual
Filtering	6	0.782772	3.7910	0.021271	0.034731
Smoothing	8	1.026919	3.5095	0.000734	0.018294

Table 1. Iteration result.

The iteration results, both for filtering and smoothing models, are shown in **Table 1**. The final cost of the smoothing model is the least compared to the final cost of the filtering model. When the trace matrix terms are considered in the cost function, the total final cost of the smoothing model is 0.019188 unit, while the total final cost of the filtering model is 0.039725 unit. The value of the trace matrix terms is 0.0185 unit. It is noticed that the output residual could be dropped to almost 52% from the filtering output residual by using the approach proposed in

this chapter. This statement is valid since the output residual of smoothing model is least than the output residual of filtering model.

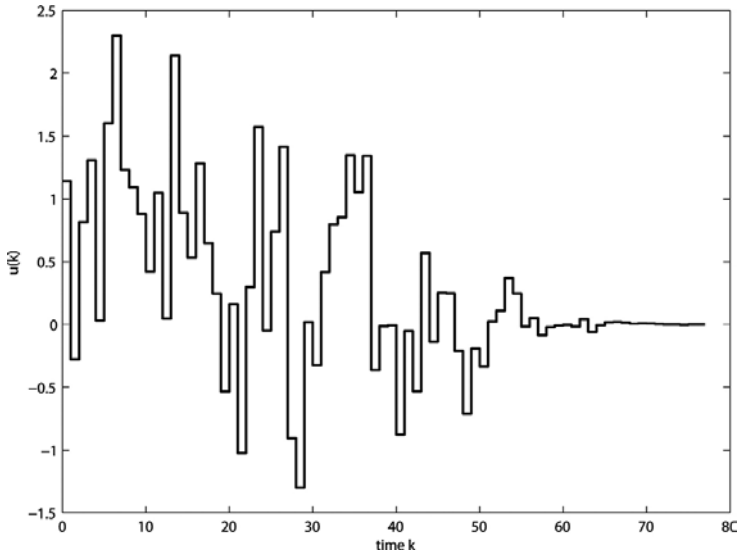

Figure 2. Filtering trajectory for final control.

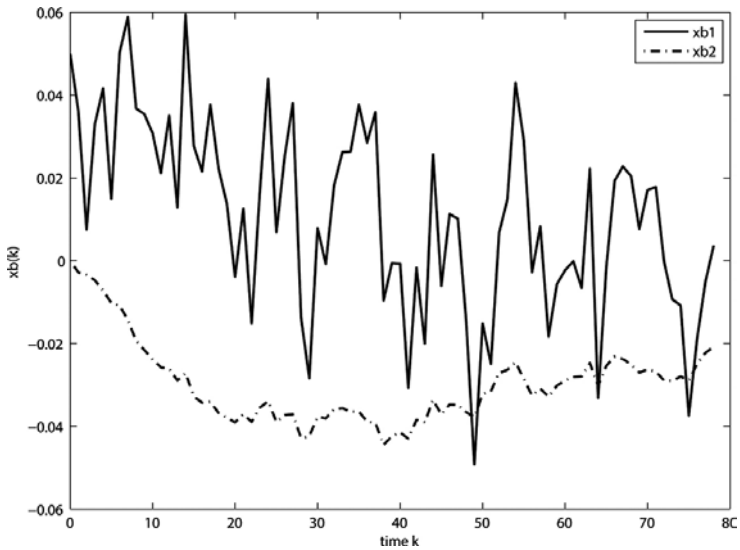

Figure 3. Filtering trajectory for final state.

To identify the accuracy of the resulting algorithm, the norms of the differences between the real plant and the model used at the end of iteration, which are 0.0128 unit for filtering model

http://dx.doi.org/10.5772/64564

and 0.0099 unit for smoothing model, are calculated. These values show that the smoothing model can approximate closely to the correct optimal solution of the original optimal control problem rather than the filtering model. Hence, the accuracy of the smoothing model is proven.

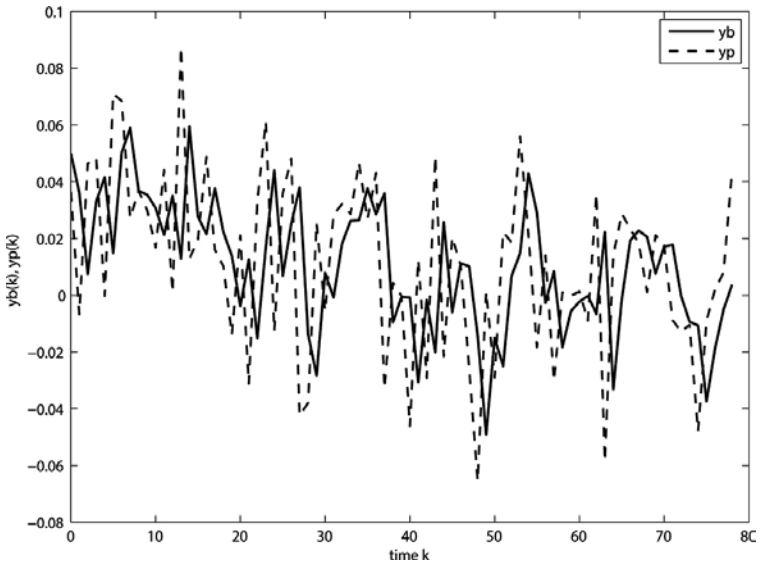

Figure 4. Filtering trajectory for final output and real output.

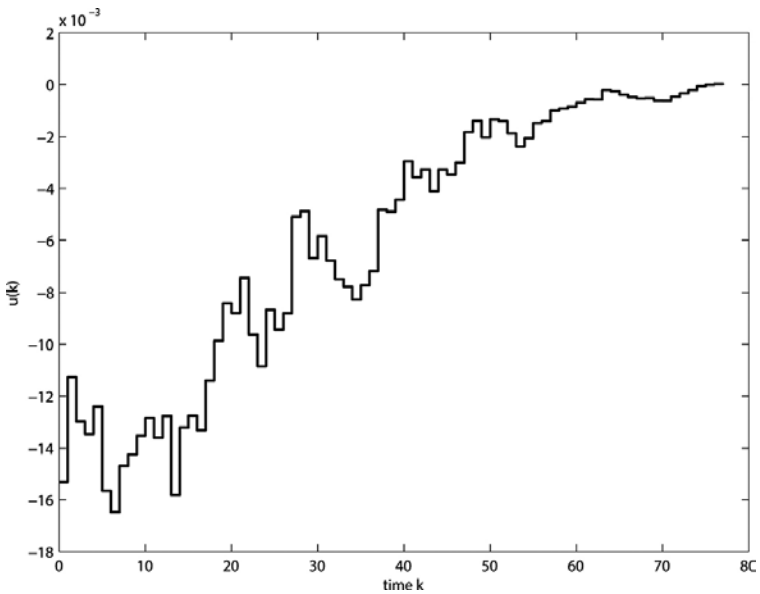

Figure 5. Smoothing trajectory for final control.

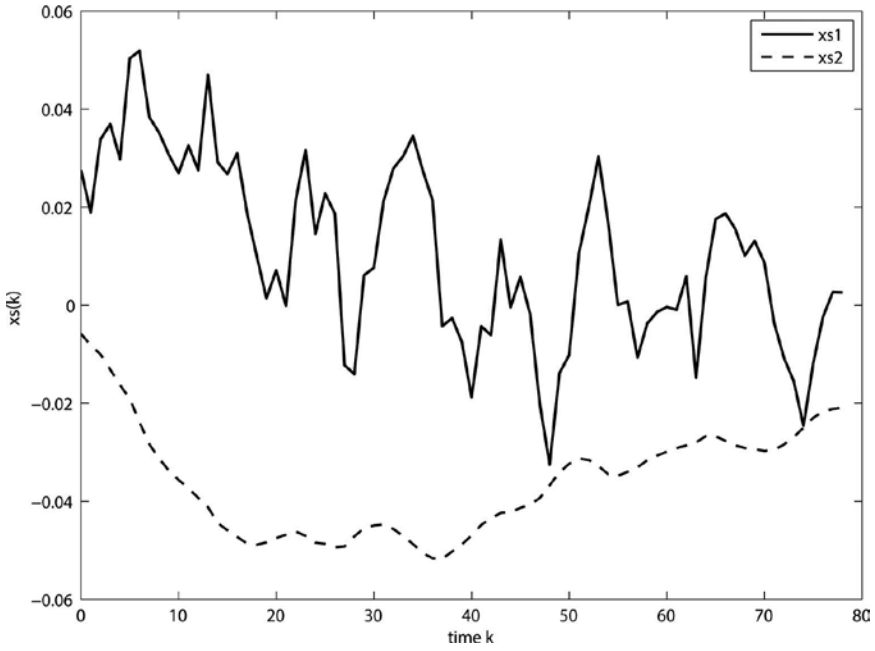

Figure 6. Smoothing trajectory for final state.

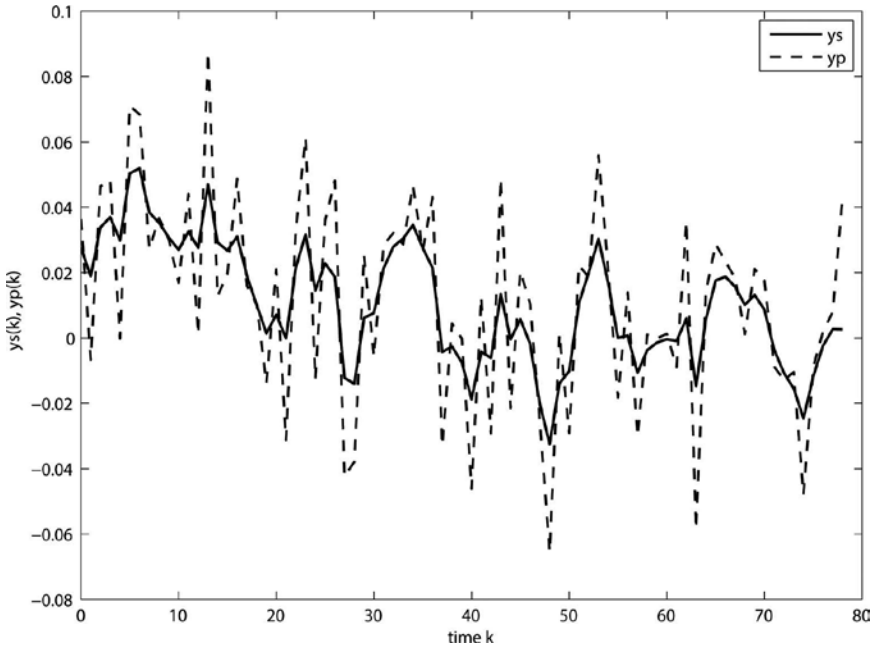

Figure 7. Smoothing trajectory for final output and real output.

http://dx.doi.org/10.5772/64564

The trajectories of final control, final state and final output for filtering, and smoothing models are shown in **Figures 2–7**. With the smallest output residual, the output, which is associated with the smoothed state estimate, is definitely applicable to measure the real output trajectory.

6. Concluding remarks

A fixed-interval smoothing scheme was modified in this chapter for solving the discrete-time nonlinear stochastic optimal control problem. The state estimation procedure, which is using the Kalman filtering theory and is followed by the fixed-interval smoothing, is applied to estimate the system dynamics. Then, the smoothed state estimate is used in designing the feedback optimal control law. By employing this smoothed state estimate, system optimization and parameter estimation are integrated. During the computation procedure, the differences between the real plant and the model used are calculated iteratively. On the other hand, the output measured from the real plant is fed back into the model used, in turn, updates the iterative solution. Once the convergence is achieved, the iterative solution approaches to the correct optimal solution of the original optimal control problem, in spite of model-reality differences. The illustrative example on the optimal control of the continuous stirred-tank reactor problem was studied. The results obtained demonstrated the applicable of the approach proposed, and the efficiency of the approach proposed is highly presented.

Acknowledgements

The authors like to thank the Universiti Tun Hussein Onn Malaysia (UTHM) for financial supporting to this study under Incentive Grant Scheme for Publication (IGSP) VOT. U417.

Author details

Sie Long Kek[1*], Kok Lay Teo[2] and Mohd Ismail Abd Aziz[3]

*Address all correspondence to: slkek@uthm.edu.my

1 Center for Research in Computational Mathematics, Universiti Tun Hussein Onn Malaysia, Parit Raja, Malaysia

2 Department of Mathematics and Statistics, Curtin University of Technology, Perth, WA, Australia

3 Department of Mathematical Sciences, Universiti Teknologi Malaysia, UTM, Skudai, Malaysia

References

[1] Kalman R. E. A new approach to linear filtering and prediction problems. Journal of Basic Engineering. 1960; 82(1):35–45.

[2] Bryson A. E. and Ho Y. C. Applied Optimal Control. Washington: Hemisphere; 1975.

[3] Bertsekas D. P. Dynamic Programming and Optimal Control (Vol. 1, No. 2). Belmont: Athena Scientific; 1995.

[4] Lewis F. L. and Syrmos V. L. Optimal Control. 2nd ed. USA: John Wiley & Sons; 1995.

[5] Feng Z.G. and Teo K. L. Optimal feedback control for stochastic impulsive linear systems subject to Poisson processes. In: Optimization and Optimal Control. New York: Springer; 2010. p. 241–258.

[6] Misiran M., Wu C., Lu Z. and Teo K.L. Optimal filtering of linear system driven by fractional Brownian motion. Dynamic Systems and Applications. 2010; 19(3):495–514.

[7] Loxton R., Lin Q. and Teo K. L. A stochastic fleet composition problem. Computers & Operations Research. 2012; 39(12):3177–3183. DOI: 10.1016/j.cor.2012.04.004.

[8] Liu C. M., Feng Z. G. and Teo K. L. On a class of stochastic impulsive optimal parameter selection problems. International Journal of Innovation, Computer and Information Control. 2009; 5:1043–1054.

[9] Yin Y., Shi P., Liu F. and Teo K. L. Robust $L_2 – L_\infty$ filtering for a class of dynamical systems with nonhomogeneous Markov jump process. International Journal of Systems Science. 2015; 46(4):599–608. DOI: 10.1080/00207721.2013.792976.

[10] Moura S. J., Fathy H. K., Callaway D. S. and Stein J. L. A stochastic optimal control approach for power management in plug-in hybrid electric vehicles. IEEE Transactions on Control Systems Technology. 2011; 19(3):545–555. DOI: 10.1109/TCST.2010.2043736.

[11] Wiegerinck W. Broek B. V. D. and Kappen H. Stochastic optimal control in continuous space-time multi-agent systems. Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI'06), Arlington, Virginia. 2006; 528–535.

[12] Zhu Y. Uncertain optimal control with application to portfolio selection model. International Journal of Cybernetics and Systems. 2010; 41(7):535–547. DOI: 10.1080/01969722.2010.511552.

[13] Hać A. Suspension optimization of a 2-DOF vehicle model using a stochastic optimal control technique. Journal of Sound and Vibration. 1985; 100(3):343–357. DOI: 10.1016/0022-460X(85)90291-3.

[14] Todorov E. Stochastic optimal control and estimation methods adapted to the noise characteristics of the sensorimotor system. Neural Computation. 2005; 17(5):1084–1108.

http://dx.doi.org/10.5772/64564

[15] Sethi S. P. Deterministic and stochastic optimization of a dynamic advertising model. Optimal Control Applications and Methods. 1983; 4(2):179–184. DOI: 10.1002/oca.4660040207.

[16] Kek S. L., Teo K. L. and Mohd Ismail A. A. An integrated optimal control algorithm for discrete-time nonlinear stochastic system. International Journal of Control. 2010; 83:2536–2545. DOI: 10.1080/00207179.2010.531766.

[17] Kek S. L., Teo K. L. and Mohd Ismail A. A. Filtering solution of nonlinear stochastic optimal control problem in discrete-time with model-reality differences. Numerical Algebra, Control and Optimization. 2012; 2(1):207–222. DOI: 10.3934/naco.2012.2.207.

[18] Kek S. L., Mohd Ismail A. A., Teo K. L. and Ahmad R. An iterative algorithm based on model-reality differences for discrete-time nonlinear stochastic optimal control problems. Numerical Algebra, Control and Optimization. 2013; 3(1):109–125. DOI: 10.3934/naco.2013.3.109.

[19] Kirk D. E. Optimal Control Theory: An Introduction. Mineola, New York: Dover Publications; 2004.

Chapter 5

Design, Analysis, and Applications of Iterative Methods for Solving Nonlinear Systems

Alicia Cordero, Juan R. Torregrosa and
Maria P. Vassileva

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64106

Abstract

In this chapter, we present an overview of some multipoint iterative methods for solving nonlinear systems obtained by using different techniques such as composition of known methods, weight function procedure, and pseudo-composition, etc. The dynamical study of these iterative schemes provides us valuable information about their stability and reliability. A numerical test on a specific problem coming from chemistry is performed to compare the described methods with classical ones and to confirm the theoretical results.

Keywords: system of nonlinear equations, iterative methods, order of convergence, weight function procedure, stability, basin of attraction

1. Introduction

The problem of solving equations and systems of nonlinear equations is among the most important in theory and practice, not only of applied mathematics, but also in many branches of science, engineering, physics, computer science, astronomy, finance, etc. A glance at the literature shows a high level of contemporary interest.

The search for solutions of systems of nonlinear equations is an old, frequent, and important problem for many applications in mathematics and engineering (for example, see Refs. [1–5]).

The main goal of this chapter is to describe different methods for approximating a solution ξ of a system of nonlinear equations $F(x) = 0$, where $F : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$ is a sufficiently differentiable function on the convex set $\Omega \subseteq \mathbb{R}^n$. The most commonly used techniques are iterative methods,

which, from an initial guess a sequence of iterates is built, were converging to the solution of the problem under some conditions. Although not as many as in the case of scalar equations, some publications have appeared in the recent years, proposing different iterative methods for solving nonlinear systems (see, for example, Refs. [6–8], among others). They have made several modifications to the classical methods to accelerate the convergence and to reduce the number of operations and functional evaluations per step of the iterative method. Newton's method is the most used iterative technique for solving these kind of problems (see Ref. [9]), whose iterative expression is

$$x^{(k+1)} = x^{(k)} - [F'(x^{(k)})]^{-1} F(x^{(k)}), k = 0,1,\ldots \tag{1}$$

where $F'(x^{(k)})$ denotes the Jacobian matrix associated to function F on $x^{(k)}$.

We remember the concepts of order of convergence and efficiency index of an iterative scheme.

Definition 1.1. *Let $\{x^{(k)}\}_{k\geq 0}$ be a sequence in $\mathbb{R}^n$ convergent to ξ. Then, the convergence is said to be linear, if there exist M, $0 < M < 1$, and $k_0 \in \mathbb{N}$ such that $\|x^{(k+1)} - \xi\| \leq M\|x^{(k)} - \xi\|$, $\forall k \geq k_0$, and of order p, $p > 1$, if there exist M, $M > 0$, and $k_0 \in \mathbb{N}$ such that $\|x^{(k+1)} - \xi\| \leq M\|x^{(k)} - \xi\|^p$, $\forall k \geq k_0$.*

On the other hand, Ostrowski [10] introduced the *efficiency index* of an iterative method as $p^{1/d}$, where p is the order of convergence and d is the number of functional evaluations per iteration. In the multidimensional case, it is more useful for the efficiency index to be defined as $p^{1/(d+op)}$, where *op* is the number of products-quotients per iteration.

The most direct technique is to adapt the methods designed for solving nonlinear equations to the multidimensional case. This process is easy only if, in the denominators of the iterative expression, does not appear any evaluation of the nonlinear function that describes the system. This is the case of Newton's schemes and Newton-type methods coming from quadrature formulas, such as those described in Refs. [11–18]. In Refs. [19] and [20], the authors designed a general procedure called *pseudo-composition* that allows to obtain predictor-corrector methods with high order of convergence. These multipoint schemes use any method as a predictor and a corrector step, where Gaussian quadrature is used.

On the other hand, other multipoint schemes have been developed by using different techniques: Adomian decomposition [21–23], the replacement of the second derivative in one-point schemes by some approximation that yields multipoint iterative methods [8, 9], the Steffensen-type methods adapted to multidimensional case (see, for instance, Refs. [17, 24], among others). In all these papers, the references therein are also important.

A generally used technique for constructing iterative schemes is the composition of known methods. This technique was introduced by Traub [9]: if two iterative methods with orders of convergence p_1 and p_2, respectively, are composed, the resulting scheme has order p_1p_2. The use of this technique greatly increases the number of functional evaluations of F and F'. Therefore,

http://dx.doi.org/10.5772/64106

it is necessary to avoid as much as possible such new evaluations by means of approximation techniques. In Ref. [25], the authors composed twice Newton's scheme with itself 'frozening' the derivatives and, by means of undetermined coefficients method, obtained the following fifth-order iterative scheme

$$z^{(k)} = y^{(k)} - 5[F'(x^{(k)})]^{-1}F(y^{(k)}),$$
$$x^{(k+1)} = z^{(k)} + \frac{1}{5}[F'(x^{(k)})]^{-1}[15F(y^{(k)}) - F(z^{(k)})],$$

where $y^{(k)}$ is a Newton's step. Let us observe that it only needs three functional evaluations and one Jacobian evaluation. Moreover, all the linear systems involved have the same matrix of coefficients. As a consequence, the efficiency index of this method is the best one, as far as we know. A similar procedure is used by Cordero et al. [26], getting Newton-Jarratt type methods of fifth and sixth orders. In addition, the following method described by Arroyo et al. [27] belongs to the class of Jarratt-type methods, but it has order of convergence five,

$$x^{(k+1)} = y^{(k)} + [F'(x^{(k)}) - 5F'(y^{(k)})]^{-1}[F'(x^{(k)}) - 5F'(y^{(k)})][F'(x^{(k)})]^{-1}F(y^{(k)}),$$

where $y^{(k)}$ is a Newton's step.

In recent years, the technique of weight functions has also been developed, mainly for scalar equations. Weight functions are introduced in the iterative scheme to increase the order of convergence without increasing the number of functional evaluations. Among others, Sharma et al. [28] constructed a fourth-order scheme by using this procedure and, more recently, Artidiello et al. [7, 29] presented different families of high-order iterative methods by using matrix weight functions.

As we have previously mentioned, most of the iterative methods for nonlinear equations are not directly extendable to systems. However, in Refs. [6, 30], the authors present a general procedure to transform any scalar iterative method to the multidimensional case.

On the other hand, the dynamical analysis of an iterative method is becoming a trend in recent publications on iterative methods for scalar equations because it allows us to classify the different iterative formulas, not only from the point of view of its order of convergence, but also analyzing how these formulas behave as a function of the initial estimate that is taken. Another advantage of this analysis is to select the more stable elements of a parametric family whose members have the same order of convergence (see, for example, Ref. [31]). A first step in this direction on nonlinear systems was given by Cordero et al. [32], and a deeper analysis was made by Cordero et al. [33], studying the behavior of several methods on particular polynomial systems. In any case, the dynamical study in the multidimensional case is an emerging research topic with a promising future.

The rest of the chapter is organized as follows. In Section 2, we describe different techniques for designing iterative methods for nonlinear systems. In Section 3, we give some touches about

the dynamic study of an iterative method for the scalar case and its extension to the multidimensional case. In order to check the introduced methods and compare them with other classical ones, in Section 4, we apply them on different test problems.

2. Design of the methods

In this section, we introduce three techniques for designing iterative methods for solving nonlinear systems of equations: pseudo-composition, weight function procedure, and a technique for extending scalar methods to the multidimensional case, in a non-trivial way.

The convergence results are going to be demonstrated by means of the n-dimensional Taylor expansion of the functions involved. Let $F: \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be a sufficiently Frèchet differentiable function in Ω. By using the notation introduced by Cordero et al. [26], the qth derivative of F at $u \in \mathbb{R}^n, q \geq 1$, is the q-linear function $F^{(q)}(u): \mathbb{R}^n \times \mathbb{R}^n \cdots \times \mathbb{R}^n \to \mathbb{R}^n$ such that $F^{(q)}(u)\left(v_1, ..., v_q\right) \in \mathbb{R}^n$. It is easy to observe that: and $F^{(q)}(u)(v_1,\ldots,v_{q-1},\cdot) \in \mathcal{L}(\mathbb{R}^n)$ and $F^{(q)}(u)\left(v_{\sigma 1}, ..., v_{\sigma q}\right) = F^{(q)}(u)\left(v_1, ..., v_q\right)$, for all permutation σ of {1, 2, …, q}. We will use the notation:

$$F^{(q)}(u)(v_1, v_2, \ldots, v_q) = F^{(q)}(u)v_1v_2\ldots v_q, \quad \text{and}$$
$$F^{(q)}(u)v^{q-1}F^{(p)}v^{(p)} = F^{(q)}(u)F^{(p)}(u)v^{q+p-1}.$$

For $\xi + h \in \mathbb{R}^n$ lying in a neighborhood of a solution ξ of the nonlinear system $F(x) = 0$ and assuming that the Jacobian matrix $F'(\xi)$ is nonsingular, Taylor's expansion can be applied, obtaining

$$F(\xi + h) = F'(\xi)\left[h + \sum_{q=1}^{p-1} C_q h^q\right] + O[h^p],$$

where $C_q = \left(\frac{1}{q!}\right)[F'(\xi)]^{-1}F^{(q)}(\xi), q \geq 2$ We observe that $C_q F^q \in \mathbb{R}^n$ since $F(q)(\xi) \in \mathcal{L}(\mathbb{R}^n \times \cdots \times \mathbb{R}^n \to \mathbb{R}^n)$. In addition, we can express the Jacobian matrix of F, F' as

$$F'(\xi + h) = F'(\xi)\left[I + \sum_{q=1}^{p-1} qC_q h^{q-1}\right] + O[h^p],$$

where I is the identity matrix. Therefore, $qC_q h^{q-1} \in \mathcal{L}(\mathbb{R}^n)$. From this expansion we can conjecture that

http://dx.doi.org/10.5772/64106

$$[F'(\xi+h)]^{-1} = [I + X_2 h + X_3 h^2 + X_4 h^3 + \cdots][F'(\xi)]^{-1} + O[h^p]$$

and taking into account that $[F'(\xi+h)]^{-1}F'(\xi+h) = F'(\xi+h)[F'(\xi+h)]^{-1} = I$, we obtain

$$X_2 = -2C_2, X_3 = 4C_2^2 - 3C_3, X_4 = -8C_2^3 + 6C_2C_3 + 6C_3C_2 - 4C_4, \ldots$$

We denote $e_k = x^{(k)} - \xi$ the error in the kth iteration. The equation $e_{k+1} = Le_k^p + O\left[e_k^{p+1}\right]$, where L is a p-linear function $L \in \mathcal{L}(\mathbb{R}^n \times \ldots \times \mathbb{R}^n, \mathbb{R}^n)$, is called *error equation*, and p is the *order of convergence*.

2.1. Pseudo-composition technique

We use the generic formulas of the Gaussian quadrature and develop families of predictor-corrector iterative methods, variants of Newton's scheme, for solving nonlinear systems. Starting with any method of order p as a predictor and correcting over Gaussian quadrature, we will show that the final order of the obtained method will depend, among other things, on the order of the last two steps of the predictor. Let

$$y^{(k)} = \xi + \sum_{j=q}^{5q-1} M_j e_k^j + O[e_k^{5q}], \quad z^{(k)} = \xi + \sum_{j=p}^{5p-1} N_j e_k^j + O[e_k^{5p}],$$

be the penultimate and last steps of any iterative method with orders of convergence p and q, respectively. Taking this scheme as a predictor, we introduce the Gaussian quadrature as a corrector and we get four cases with the following iterative formulas:

$$\begin{aligned} &\text{(a)} \quad x^{(k+1)} = y^{(k)} - 2K^{-1}F(y^{(k)}), \qquad \text{(b)} \quad x^{(k+1)} = z^{(k)} - 2K^{-1}F(z^{(k)}), \\ &\text{(c)} \quad x^{(k+1)} = y^{(k)} - 2K^{-1}F(z^{(k)}), \qquad \text{(d)} \quad x^{(k+1)} = z^{(k)} - 2K^{-1}F(y^{(k)}), \end{aligned} \tag{2}$$

where for all cases $K = \sum_{i=1}^{m} \omega_i F'(\eta_i^{(k)})$, $\eta_i^{(k)} = \frac{1}{2}[(1+\tau_i)z^{(k)} + (1-\tau_i)y^{(k)}]$, and ω_i and τ_i are the weights and nodes, respectively, of the orthogonal polynomial of degree m, which defines the corresponding Gaussian quadrature. Then, $\eta_i^{(k)}$ is calculated by using the points obtained in the last two steps of the predictor.

To simplify the calculations, we use the following notation: $\sum_{j=q}^{5q-1} M_j e_k^j = A_{1(q)}$ and $\sum_{j=q}^{5q-1} N_j e_k^j = A_{2(p)}$, where the subscripts in parentheses denote the value of the smallest power assumed by j in the sum. By using this notation, $\eta_i^{(k)}$ can be expressed as $\eta_i^{(k)} = \xi + \frac{1}{2}(R + \tau_i S)_{(q)}$, where $R = A_{2(p)} + A_{1(q)}$ and $S = A_{2(p)} - A_{1(q)}$. By expansion of $F(y^{(k)}), F(z^{(k)})$ and $F'(\eta_i^{(k)})$ in the Taylor series around ξ, we obtain

$$F(y^{(k)}) = F'(\xi)[A_{1(q)} + C_2 A_{1(2q)}^2 + C_3 A_{1(3q)}^3 + C_4 A_{1(4q)}^4] + O[e_k^{5q}],$$
$$F(z^{(k)}) = F'(\xi)[A_{1(p)} + C_2 A_{1(2p)}^2 + C_3 A_{1(3p)}^3 + C_4 A_{1(4p)}^4] + O[e_k^{5p}],$$
$$F(\eta_i^{(k)}) = F'(\xi)[I + B + C\tau_i + D\tau_i^2 + E\tau_i^3] + O[e_k^{4q}],$$

where $B = C_1 R + \frac{3}{4} C_3 R^2 + \frac{1}{2} C_4 R^3, C = C_2 S + \frac{3}{4} C_3 (RS + SR) + \frac{1}{2} C_4 (R^2 S + RSR + SR^2)$, $D = \frac{3}{4} C_3 S^2 + \frac{1}{2} C_4 (SRS + S^2 R + R^2 S)$ and $E = \frac{1}{2} C_4 S^3$. We also introduce the following notation: $\sum_{i=1}^{m} \omega_i = \sigma$ and $\frac{1}{\sigma} \sum_{i=1}^{m} \omega_i \tau_i^j = \sigma_j$ with $j = 1, 2, \ldots$, which will allow us to simplify the analysis of the convergence conditions of the described methods.

Now, we develop the expression $K = \sum_{i=1}^{m} \omega_i F'(\eta_i^{(k)})$ appearing in Eq. (2) and we obtain $K = \sigma F'(\xi)\left[I + K_{1(q)} + K_{2(2q)} + K_{3(3q)}\right] + O[e_k^{4q}]$, where $K_{1(q)} = C_2 (R + \sigma_1 S)_{(q)}$, $K_{2(2q)} = \frac{3}{4} C_3 (R^2 + \sigma_1 (RS + SR) + \sigma_2 S^2)_{(2q)}$ and $K_{3(3q)} = [\frac{1}{2} C_4 (R^3 + \sigma_1 \left(R^2 S + RSR + SR^2\right) + \sigma_2 \left(RS^2 + SRS + S^2 R\right) + \sigma_2 S^3)_{(3q)}$. Recalling that $K^{-1} K = I$, we get $K^{-1} = \sigma^{-1}\left[I + K'_{1(q)} + K'_{2(2q)} + K'_{3(3q)}\right][F'(\xi)]^{-1} + O[e_k^{4q}]$, where $K'_{1(q)} = -K_{1(q)}, K'_{2(2q)} = (K_1^2 - K_2)_{(2q)}$, and $K'_{3(3q)} = \left(-K_1^3 + K_2 K_1 + K_1 K_2 - K_3\right)_{(3q)}$. Therefore, considering case (a), we obtain for $L = 2K^{-1} F(y^{(k)})$ the following expression:

$$L = \frac{2}{\sigma} A_{1(q)} + \frac{2}{\sigma}[(C_2 A_1 + K'_1) A_1]_{(2q)} + \frac{2}{\sigma}[(C_2 A_1^2 + K'_1 C_2 A_1 + K'_2) A_1]_{(3q)}$$
$$+ \frac{2}{\sigma}[(C_4 A_1^3 + K'_1 C_3 A_1^2 + K'_2 C_2 A_1 + K'_3) A_1]_{(4q)} + O[e_k^{4q}].$$

Since $x^{(k+1)} = y^{(k)} - L$, the error equation can be expressed as

http://dx.doi.org/10.5772/64106

$$e_{k+1} = A_{1(q)} - \frac{2}{\sigma} A_{1(q)} - \frac{2}{\sigma}[(C_2 A_1 + K'_1) A_1]_{(2q)} - \frac{2}{\sigma}[(C_2 A_1^2 + K'_1 C_2 A_1 + K'_2) A_1]_{(3q)}$$
$$- \frac{2}{\sigma}[(C_4 A_1^3 + K'_1 C_3 A_1^2 + K'_2 C_2 A_1 + K'_3) A_1]_{(4q)} + O[e_k^{4q}]$$

We note that if $\sigma = 2$ we obtain order of convergence at least $2q$. The possibility of obtaining a convergence order greater than $2q$ depends on the expression $(C_2 A_1 + K'_1) A_1$. We develop it and get $\left(C_2 A_1 + K'_1\right) A_1 = \sigma_1 C_2 \left(A_1^2\right)_{(2q)} - (1 + \sigma_1) C_2 \left(A_2 A_1\right)_{(p+q)}$. Then, $\sigma_1 = 0$, the error equation is:

$$e_{k+1} = C_2 (A_2 A_1)_{(p+q)} - \frac{2}{\sigma}[(C_2 A_1^2 + K'_1 C_2 A_1 + K'_2) A_1]_{(3q)}$$
$$- \frac{2}{\sigma}[(C_4 A_1^3 + K'_1 C_3 A_1^2 + K'_2 C_2 A_1 + K'_3) A_1]_{(4q)} + O[e_k^{4q}]. \tag{3}$$

It is clear that the higher order of convergence in case (a) is min{p+q, 3q}. Finally, it is easy to prove that in case (b), the order of convergence of the resulting method is at least p. If $\sigma = 2$ the convergence order is $p + q$ unless $\sigma_1 = 2$ or $C_2 = 0$, being in case the order of convergence is at least $2q + p$. In cases (c) and (d) the convergence order is q, which is lower than the order of the predictor. It should be noticed that in case (b), two further functional evaluations are required and a new linear system per step must be solved. This causes the obtained method to be inefficient, from the standpoint of computational efficiency. In addition, we would like to remark that the order of convergence can be greater than $p + q$ depending on expressions A_1 and A_2, which represent the errors of penultimate and last steps of the predictor method, and also of σ_1 in case (b). The comments above allow us to state the following result, whose proof can be found by Cordero et al. [19], which establishes the order of convergence of the schemes that are obtained by using any method as a predictor of order p and eventually correcting it by the use of Gaussian quadrature, in case (a).

Theorem 2.1. *Let $F: \Omega \subset \mathbb{R}^n \to \mathbb{R}^n$ be sufficiently differentiable function in Ω and $F'(x)$ continuous and nonsingular at $\xi \in \Omega$, solution of the nonlinear system. Let $y^{(k)}$ and $z^{(k)}$ be the penultimate and last steps of orders q and p, respectively, of a certain iterative method. Taking this scheme as a predictor we get a new approximation $x^{(k+1)}$ of ξ given by Eq. (2). Then,*

i. *the methods of the obtained families have an order of convergence at least q,*

ii. *if $\sigma = 2$ is satisfied, then the order of convergence is at least 2q,*

iii. *if, also, $\sigma_1 = 0$ the order of convergence is min{p + q, 3q}.*

In **Table 1**, we show the values of σ and σ_1 for some Gaussian quadrature.

In terms of computational efficiency, the most efficient methods are those which use fewer nodes and few functional evaluations, so we only consider case (a). Also, Theorem 2.1 shows that the order of convergence does not depend on the number of nodes, it only depends on the order of convergence of the penultimate and last step of the predictor method. Therefore, it is computationally more efficient to use one or two nodes. We note that the Gauss-Chebyshev quadrature does not fulfill the second condition of Theorem 2.1. Then, its order of convergence is q. The method obtained by using Gauss-Radau quadrature of one node does not fulfill the third condition but it verifies the second one; hence, its order of convergence is at least $2q$. The remaining quadrature with nodes 1, 2, and 3 satisfies the conditions of Theorem 2.1 and the order of convergence is at least $\min\{p + q, 3q\}$.

Number of nodes	Gaussian Quadrature							
	Chebyshev		Legendre		Lobatto		Radau	
	σ	σ_1	σ	σ_1	σ	σ_1	σ	σ_1
1	π	0	2	0	2	0	2	−1
2	π	0	2	0	2	0	2	0
3	π	0	2	0	2	0	2	0

Table 1. Quadrature formula used.

If we use case (a) and the Gauss-Legendre quadrature with 1 node or Gauss-Lobatto quadrature with one node such as corrector, we obtain the midpoint method, where $K = 2F'\left(\frac{y^{(k)} + z^{(k)}}{2}\right)$. In case of using Gauss-Radau quadrature with one node, we obtain Newton's method ($K = F'(y^{(k)})$). Finally, if we use Gauss-Lobatto quadrature with two nodes or Gauss-Radau quadrature with two nodes such as corrector, we obtain trapezoidal method with $K = 2F'\left(y^{(k)} + z^{(k)}\right)$ and Noor's scheme where $K = 8\left[F'\left(y^{(k)}\right) + 3F'\left(\frac{y^{(k)} + 2z^{(k)}}{3}\right)\right]$, respectively.

2.2. Weight function procedure

The different methods obtained in the previous section are not optimal in the sense of Kung-Traub conjecture [34], when they are applied to scalar equations. By using the weight functions technique, we can increase the order of convergence of the designed methods without adding new functional evaluations.

We denote by $X = \mathbb{R}^{n \times n}$ the Banach space of all $n \times n$ real square matrices. The weight function in this context is a Frèchet differentiable function $H: X \to X$ satisfying:

(i) $H'(u)(v) = H_1 uv$, H' being the first derivative of H, $H': X \to \mathcal{L}(X)$, $H_1 \in \mathbb{R}$ and $\mathcal{L}(X)$ denotes the space of linear mapping from X to itself.

http://dx.doi.org/10.5772/64106

(ii) $H''(u,v)(w) = H_2uvw$, H'' being the second derivative of $H, H'': X \times X \to \mathcal{L}(X), H_2 \in \mathbb{R}$.

Then, the Taylor expansion of H around the identity matrix I, of size $n \times n$, gives

$$H(\psi^{(k)}) \approx H_0(I) + H_1(\psi^{(k)} - I) + \frac{1}{2}H_2(\psi^{(k)} - I)^2$$

Now, by using a relaxed Newton's method as a predictor and a weight function procedure in the corrector step, we design the following families of two-point schemes:

$$\begin{aligned} z^{(k)} &= x^{(k)} - \beta[F'(x^k)]^{-1}F(x^k), \\ x^{(k+1)} &= z^{(k)} - 2H(u^{(k)})[\sum_{n}^{i=1}\omega_i F'(\eta_i^{(k)})]^{-1}F(x^{(k)}) \end{aligned} \tag{4}$$

where β is a non-zero real parameter and $u^{(k)} = \frac{1}{\sum_{i=1}^{m}\omega_i}\left[F'(x^{(k)})\right]^{-1}\sum_{i=1}^{m}\omega_i F'(\eta_i^{(k)})$. Here, we have used the same notations previously introduced. The following result establishes the order of convergence of this iterative scheme under some conditions of function H.

Theorem 2.2. *Let $\xi \in \Omega$ be a zero of a sufficiently differentiable function $F: \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$. Let us also suppose that the initial estimation $x^{(0)}$ is close enough to the solution ξ and $F'(\xi)$ is nonsingular. The iterative methods (Eq. 4) have order of convergence four if*

$$\beta = \frac{4(1+\sigma_1)}{3(1+2\sigma_1+\sigma_2)}$$

and a sufficiently differentiable function H is chosen satisfying the conditions

$$H_0(I) = \frac{\sigma}{2}I, \quad H_1 = \frac{\sigma(1+2\sigma_1+4\sigma_1^2+3\sigma_2)}{8(1+\sigma_1)^2}$$
$$H_2 = -\frac{3\sigma(-1-4\sigma_1-2\sigma_1^2+4\sigma_1^3-4\sigma_2-8\sigma_1\sigma_2+2\sigma_1^2\sigma_2-3\sigma_2^2)}{8(1+\sigma_1)^4}$$

and H''' is a bounded operator, where σ and σ_j $j = 1, 2$, depend on the Gaussian quadrature used (see Ref. [35]).

Table 2 shows the values of σ_1, σ_2, and β, depending on the weights and the nodes of the orthogonal polynomials used in the Gaussian quadrature and also the corresponding values of the weight function and its derivatives, which are used in the iterative scheme.

Quadrature	No. of nodes	σ	σ_1	σ_2	β	$H_0(I)$	H_1	H_2
Gauss-Chebyshev	*1*	π	*0*	*0*	*4/3*	$(\pi/2)I$	$\pi/8$	$3\pi/8$
Gauss-Legendre	*1*	*2*	*0*	*0*	*4/3*	*I*	*1/4*	*3/4*
Gauss-Lobatto	*1*	*2*	*0*	*0*	*4/3*	*I*	*1/4*	*3/4*
	2	*2*	*0*	*1*	*2/3*	*I*	*−1/2*	*6*
Gauss-Radau	*1*	*2*	*−1*	*0*	*0*	*(1/2)I*	–	–
	2	2	0	1/3	1	I	0	2

Table 2. Nodes, $H_0(I)$, H_1 and H_2, for different Gaussian quadrature.

Let us remark that the class of methods designed allows the use of any Gaussian quadrature. In fact, when the technique of pseudo-composition was defined in Section 2.1 based also in Gaussian quadrature, the Chebyshev orthogonal polynomials could not be employed, as they did not verify the hypothesis of the main theorem. Nevertheless, with this new design, all the orthogonal polynomials can be applied and all of them derive optimal methods in the scalar case. In practice, we only use quadrature formulas with one and two nodes because, according to Theorem 2.2, the order of convergence is independent of the number of nodes.

For the one-dimensional case, according to the Kung-Traub's conjecture [34], the obtained fourth-order methods are optimal (in case of Gauss-Radau with one node, classical Newton's method is obtained). Other new methods are obtained in the rest of cases.

Quadrature	No. of nodes	Name	$H(t)$
Gauss-Chebyshev	1	GC1	$\frac{\pi}{16}\frac{5-12t+15t^2}{t^2}$
Gauss-Legendre	1	GLe1	$\frac{1}{8}(9-4t+3t^2)$
Gauss-Lobatto	2	GLo2	$\frac{9}{2}-\frac{13}{2}t+3t^2$
Gauss-Radau	2	GR2	t^2-2t+2

Table 3. Notation and weight functions for different Gaussian quadrature formulas.

In **Table 3**, we show the weight functions used for each iterative scheme coming from the respective Gaussian quadrature rules. Let us note that the iterative method coming from Gauss-Lobatto with one node is the same as the one resulting from the application of Gauss-Legendre, also with one node. In this case, both coincide with the fourth-order procedure recently published by Sharma et al. [28]. Let us note that other weight functions should derive in other new schemes. For example, the expression of the method obtained by using Gauss-Legendre quadrature with one node is

http://dx.doi.org/10.5772/64106

$$y^{(k)} = x^{(k)} - \frac{4}{3}[F'(x^{(k)})]^{-1}F(x^{(k)}), \quad z^{(k)} = \frac{1}{2}(x^{(k)} + y^{(k)}),$$
$$x^{(k+1)} = x^{(k)} - \frac{9}{8}[F'(x^{(k)})]^{-1}F(x^{(k)}) + \left(\frac{1}{2}I - \frac{3}{8}[F'(x^{(k)})]^{-1}[F'(z^{(k)})]^{-1}\right)[F'(x^{(k)})]^{-1}F(x^{(k)}).$$

2.3. Divided difference operator

In this section, we present a design published by Cordero et al. [30] of some families of parametric iterative methods for solving nonlinear equations by means of some known schemes and afterwards extend one of them to systems of nonlinear equations. For this purpose, we use Ostrowski's [10] and Chun's [36] methods with iterative schemes $x_{k+1} = y_k - \frac{f(x_k)}{f(x_k) - 2f(y_k)}\frac{f(y_k)}{f'(x_k)}$ and $x_{k+1} = y_k - \frac{f(x_k) + 2f(y_k)}{f(x_k)}\frac{f(y_k)}{f'(x_k)}$, respectively, where y_k is a Newton's step. We propose a new family as a generalization of the previous methods in the following form:

$$\begin{aligned} y_k &= x_k - \alpha\frac{f(x_k)}{f'(x_k)} \\ x_{k+1} &= y_k - \left[\frac{f(x_k)}{a_1 f(x_k) + a_2 f(y_k)} + \frac{b_1 f(x_k) + b_2 f(y_k)}{f(x_k)}\right]\frac{f(y_k)}{f'(x_k)}, \end{aligned} \tag{5}$$

where α, a_1, a_2, b_1, and b_2 are real parameters. In the following result, we show which values of these parameters are necessary to guarantee at least order of convergence 4.

Theorem 2.3. *Let $f: I \subseteq \mathbb{R} \to \mathbb{R}$ be a sufficiently differentiable function in an open interval I, such that $\xi \in I$ is a simple root of the nonlinear equation $f(x) = 0$. If $\alpha = 1, a_2 = a_1^2(b_2 - 2), b_1 = 1 - \frac{1}{a_1}$ and for all a_1 and $b_2 \in \mathbb{R}$ with, $a_1 \neq 0$ then sequence $\{x_k\}_{k \geq 0}$ obtained from (Eq. 5) converges to ξ with local order of convergence at least four. In this case, the error equation is*

$$e_{k+1} = ((5 - a_1(b_2 - 2)^2)c_2^3 - c_2 c_3)e_k^4 + O[e_k^5],$$

where $e_k = x_k - \xi$ and $c_q = \left(\frac{1}{q!}\right)\frac{f^{(q)}(\xi)}{f'(\xi)}, q \geq 2$.

It is easy to prove this result by using any symbolic software as Wolfram Mathematica. To extend family (Eq. 5) to multivariate case, we need to rewrite its iterative expression in such a way that no functional evaluations of *f* remain at the denominator, as they will become vectors

in the multidimensional case. So, let us consider that the first step of (Eq. 5) can be rewritten as $f(x_k) = \frac{1}{\alpha}(x_k - y_k)f'(x_k)$. By using this, we can rewrite quotient $\frac{f(y_k)}{f(x_k)}$ as

$$\frac{f(y_k)}{f(x_k)} = 1 - \alpha \frac{f[x_k, y_k]}{f'(x_k)},$$

where $f[x_k, y_k] = \frac{f(x_k) - f(y_k)}{x_k - y_k}$ is the first-order divided difference. By using this transformation, the proposed family (Eq. 5) is fully extensible to several variables in the following way

$$\begin{aligned} y^{(k+1)} &= x^{(k)} - \alpha[F'(x^{(k)})]^{-1}F(x^{(k)}), \\ x^{(k+1)} &= y^{(k)} - (G_1(x^{(k)}, y^{(k)}) + G_2(x^{(k)}, y^{(k)}))[F'(x^{(k)})]^{-1}F(y^{(k)}), \\ G_1(x^{(k)}, y^{(k)}) &= [(a_1 + a_2)I - \alpha a_2[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]]^{-1}, \\ G_2(x^{(k)}, y^{(k)}) &= (b_1 + b_2)I - \alpha b_2[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F], \end{aligned} \tag{6}$$

where $[x^{(k)}, y^{(k)}; F]$ denotes the divided difference operator of F on $x^{(k)}$ and $y^{(k)}$, I is the identity matrix, and $F'(x^{(k)})$ is the Jacobian matrix of the system.

Since the analysis of the local convergence is based on Taylor series expansion around the solution, we need to obtain the corresponding development of the divided difference operator. Let us denote by $[x^{(k)}, y^{(k)}; F]$ the divided difference operator defined by Ortega and Rheinboldt [37] as the function $[\cdot, \cdot; F]: \Omega \times \Omega \in \mathbb{R}^n \times \mathbb{R}^n \to \mathcal{L}(\mathbb{R}^n)$ that satisfies $[x, y; F](x - y) = F(x) - F(y)$, $\forall x, y \in \Omega$. To achieve this, we use the Genocchi-Hermite formula (see [37])

$$[x, x + h; F] = \int_0^1 F'(x + th)dt$$

and, by developing $F'(x + th)$ in Taylor series around x, we obtain

$$\int_0^1 F'(x + th)dt = F'(x) + \frac{1}{2}F''(x)h + \frac{1}{6}F'''(x)h^2 + O[h^3]. \tag{7}$$

Assuming that

http://dx.doi.org/10.5772/64106

$$\begin{aligned} F(x) &= F'(\xi)(e_k + C_2e_k^2 + C_3e_k^3 + C_4e_k^4) + O[e_k^5], \\ F'(x) &= F'(\xi)(I + 2C_2e_k + 3C_3e_k^2 + 4C_4e_k^3) + O[e_k^4], \\ F''(x) &= F'(\xi)(2C_2 + 6C_3e_k + 12C_4e_k^2) + O[e_k^3], \\ F'''(x) &= F'(\xi)(6C_3 + 24C_4e_k) + O[e_k^2], \end{aligned} \tag{8}$$

and replacing these developments in the formula of Genocchi-Hermite, denoting the second point of the divided difference by $y = x + h$ and the error at the first step by $e_{k,y} = y - \xi$, we have $[x, y; F] = F'(\xi)\left[I + C_2(e_{k,y} - e_k) + C_3e_k^2\right] + O\left[e_k^3\right]$. In particular, if y is the approximation of the solution provided by Newton's method, i.e., $h = x - y = [F'(x)]^{-1}F(x)$, we obtain $[x, y; F] = F'(\xi)\left[I + C_2e_k + \left(C_2^2 + C_3\right)e_k^2\right] + O\left[e_k^3\right]$. These tools allow us to prove the following result.

Theorem 2.4. *Let $f : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be a sufficiently differentiable function in a convex set Ω, and $\xi \in \Omega$ be a solution of the nonlinear system of equations $F(x) = 0$. Then, the sequence $\left\{x^{(k)}\right\}_{k \geq 0}$ obtained by using expression (6) converges to ξ with local order of convergence at least four if $\alpha = 1, a_2 = a_1^2(b_2 - 2), b_1 = 1 - \frac{1}{a_1}$ and for all a_1 and $b_2 \in \mathbb{R}$ with $a_1 \neq 0$. The error equation is*

$$e_{k+1} = ((5 - a_1(b_2 - 2)^2)C_2^3 - C_2C_3)e_k^4 + O[e_k^5],$$

where $e_k = x^{(k)} - \xi$ and $C_q = \left(\frac{1}{q!}\right)[F'(\xi)]^{-1}F^{(q)}(\xi), q \geq 2$.

Proof: By using Taylor expansion around ξ, we obtain:

$$\begin{aligned} F(x^{(k)}) &= F'(\xi)(e_k + C_2e_k^2 + C_3e_k^3 + C_4e_k^4) + O[e_k^5], \\ F'(x^{(k)}) &= F'(\xi)(I + 2C_2e_k + 3C_3e_k^2 + 4C_4e_k^3) + O[e_k^4]. \end{aligned}$$

Let us consider $\left[F'\left(x^{(k)}\right)\right]^{-1} = \left(I + X_2e_k + X_3e_k^2 + X_4e_k^3\right)[F'(\xi)]^{-1} + O\left[e_k^4\right]$. Forcing $\left[F'\left(x^{(k)}\right)\right]^{-1}F'\left(x^{(k)}\right) = I$, we get $X_2 = -2C_2$, $X_3 = 2C_2^2 - 3C_3$ and $X_4 = -4C_4 + 6C_3C_2 - 4C_2^2 + 6C_2C_3$. These expressions allow us to obtain for first step of iterative formula (6)

$$y^{(k)} = \xi + (1 - \alpha)e_k - \alpha(A_2e_k^2 + A_3e_k^3 + A_4e_k^4) + O[e_k^5], \tag{9}$$

where $A_2 = -C_2 - X_2, A_3 = -C_3 - C_2X_2 - X_3$ and $A_4 = -C_4 - C_3X_2 - C_2X_3 - X_4$. By using these results and the Taylor series expansion around ξ, we obtain

$$F(y^{(k)}) = F'(\xi)(B_1e_k + B_2e_k^2 + B_3e_k^3 + B_4e_k^4) + O[e_k^5],$$

where $B_1=\beta$, $B_2=(\alpha + \beta^2)C_2$, $B_3=-\alpha A_3 + 2\alpha\beta C_2A_3 + 3\alpha\beta^2C_3C_2 + \beta^4C_4$, $B_4=-\alpha A_4 + \alpha^2C_2{}^3-2\alpha\beta C_2A_3 + 3\alpha\beta^2C_3C_2 + \beta^4C_4$ and $\beta = 1 - \alpha$. We calculate the Taylor expansion of $[x^{(k)}, y^{(k)};F]$ by using Eq. (9), $[x^{(k)}, y^{(k)};F]= F'(\xi)(I + D_2e_k + D_3e_k{}^2 + D_4e_k{}^3) + O[e_k{}^4]$, where $D_2= (2-\alpha)C_2$, $D_3= \alpha C_2{}^2 + (3-3\alpha + \alpha^2)C_3$ and $D_4= 2\alpha C_2C_3 + \alpha(3-2\alpha)C_3C_2-(4-6\alpha + 4\alpha^2-\alpha^3)C_4$.

Then,

$$M = (a_1 + a_2)I - \alpha a_2[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]= a_1 + E_2e_k + E_3e_k^2 + E_4e_k^3 + O[e_k^4],$$

where $E_2 = \alpha a_2C_2$, $E_3 = \alpha C_2{}^3 + \alpha(\alpha-3)C_3$ and $E_4 = 6\alpha C_2C_3-2\alpha C_2{}^3-4C_4 + 5\alpha(2-\alpha)C_3C_2 + (4-6\alpha + 4\alpha^2-\alpha^3)C_3C_4$.

Thus, we obtain $G_1(x^{(k)}, y^{(k)})$ as the inverse of matrix M: $G_1(x^{(k)}, y^{(k)})=I + Y_2e_k+Y_3e_k{}^2+Y_4e_k{}^3+O[e_k{}^4]$, where $Y_2 = \frac{\alpha a_2}{a_1}C_2$, $Y_3 = \frac{\alpha a_2}{a_1^2}\left[\left(\alpha a_2 - 3\right)C_2^2 + (\alpha - 3)C_3\right]$, $Y_4 = \frac{\alpha a_2}{a_1^3}\left(8a_1 + 3\alpha a_1a_2 + 3\alpha a_2 - \alpha^2a_2^3\right)C_2^3$ and $G_2(x^{(k)}, y^{(k)}) = b_1+F_2e_k+F_3e_k{}^2+F_4e_k{}^3+F_5e_k{}^4+O[e_k{}^5]$ where $F_2=\alpha b_2C_2$, $F_3=-\alpha b_2[2C_2{}^2-(\alpha-3)C_3]$ and $F_4=b_2[\alpha(6-4\alpha+\alpha^2)C_4-6\alpha(2-\alpha)C_3C_2+4(\alpha+1)C_2{}^3-6(\alpha+1)C_2C_3]$.

Finally, we obtain the error equation of the proposed method

$$e_{k+1} = H_1e_k + H_2e_k^2 + H_3e_k^3 + H_4e_k^4 + O\left[e_k^5\right],$$

where $H_1 = \frac{1}{a_1}\left(1 + a_1\left(b_1 - 1\right)\right)(\alpha - 1)$. If $\alpha = 1$, then $H_1 = 0$ and the error equation takes the form: $e_{k+1} = H_2'e_k^2 + H_3'e_k^3 + H_4'e_k^4 + O\left[e_k^5\right]$ where $H_2' = -\frac{1}{a_1}\left(1 + a_1(b_1 - 1)\right)C_2$. We note that if $b_1 = 1 - 1/a_1$, then $H_2' = 0$ We introduce this value of b_1 and obtain the new form of the error equation $e_{k+1} = H_3''e_k^3 + H_4''e_k^4 + O\left[e_k^5\right]$ where $H_3'' = \left(1/a_1^2\right)\left[a_2 - a_1^2(b_2 - 2)\right]C_2^2$. Finally, if $a_2 = a_1^2(b_2 - 2)$, the error equation is:

$$e_{k+1} = -[(a_1(b_2 - 2)^2 - 5)C_2^3 + C_2C_3]e_k^4 + O[e_k^5]$$

http://dx.doi.org/10.5772/64106

and the proof is finished.□

By using the same hypothesis as in Theorem 2.4, the iterative scheme of the class (Eq. 6) takes the form:

$$\begin{aligned} y^{(k+1)} &= x^{(k)} - \left[F'(x^{(k)})\right]^{-1} F(x^{(k)}), \\ x^{(k+1)} &= y^{(k)} - (G_1(x^{(k)}, y^{(k)}) + G_2(x^{(k)}, y^{(k)}))[F'(x^{(k)})]^{-1} F(y^{(k)}), \\ G_1(x^{(k)}, y^{(k)}) &= \frac{1}{a_1}\left[(1 + a_1 b_2 - 2a_1)I - a_1(b_2 - 2)[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]\right]^{-1} \\ G_2(x^{(k)}, y^{(k)}) &= \frac{1}{a_1}\left[(a_1 + a_1 b_2 - 1)I - b_2[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]\right]. \end{aligned} \tag{10}$$

In the following we propose some particular cases:

1. When $a_1 = 1$, the iterative expression, being $G\left(x^{(k)}, y^{(k)}\right) = G_1\left(x^{(k)}, y^{(k)}\right) + G_2\left(x^{(k)}, y^{(k)}\right)$ takes the form:

$$\begin{aligned} G(x^{(k)}, y^{(k)}) = &\left[(b_2 - 1)I - (b_2 - 2)[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]\right]^{-1} \\ &+ b_2 I - b_2[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F] \end{aligned}$$

and we have a family of schemes with interesting particular cases, among others,

a) If $b_2 = 2$, Chun's method transferred to systems is obtained

$$x^{(k+1)} = y^{(k)} - \left(I - 2\left[F'\left(x^{(k)}\right)\right]^{-1}\left[x^{(k)}, y^{(k)}; F\right]\right)\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(y^{(k)}\right).$$

b) If $b_2 = 0$ we get Ostrowski's scheme transferred to systems

$$x^{(k+1)} = y^{(k)} - \left(-I + 2\left[F'\left(x^{(k)}\right)\right]^{-1}\left[x^{(k)}, y^{(k)}; F\right]\right)^{-1}\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(y^{(k)}\right).$$

2. When $b_2 = 0$ for any $a_1 \neq 0$, the iterative expression of the parametric family is

$$G(x^{(k)}, y^{(k)}) = \frac{(3 - 2a_1)I + 2(a_1 - 1)[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]}{(1 - 2a_1)I + 2a_1[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]}.$$

If we express $-2(a_1 - 1) = \beta$ we get the King's family transferred to systems

$$x^{(k+1)} = y^{(k)} - \frac{(1+\beta)I + \beta\left[F'\left(x^{(k)}\right)\right]^{-1}\left[x^{(k)}, y^{(k)}; F\right]}{(\beta-1)I + (\beta-2)\left[F'\left(x^{(k)}\right)\right]^{-1}\left[x^{(k)}, y^{(k)}; F\right]}\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(y^{(k)}\right).$$

3. When $b_2 = 1$ for any $a_1 \neq 0$ and $a_1 \neq 1$ the iterative form, the last step of the method (Eq. 10) takes the form:

$$x^{(k+1)} = y^{(k)} - (\frac{1}{a_1}[(1-a_1)I - a_1[F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]]^{-1} + \frac{1}{a_1}[(2a_1 - 1)I - [F'(x^{(k)})]^{-1}[x^{(k)}, y^{(k)}; F]])[F'(x^{(k)})]^{-1}F(y^{(k)}).$$

3. Dynamic studies of some methods

In the last years, a new branch of the analysis of iterative methods for solving nonlinear equations or systems has taken relevance: the dynamical analysis of the rational functions associated with the fixed point operator associated with the iterative scheme on polynomials. By using complex or real dynamics techniques, the stability and reliability of a method can be checked. Indeed, if a parametric family of iterative procedures is considered, this kind of analysis allows selecting those elements of the class with better stability and also to know which ones behave chaotically even on the most simple functions, as low degree polynomials.

The use of these dynamical tools is very frequent on scalar iterative methods; see, for example, Refs. [38–45] and the references therein, but in the multidimensional case, it is a starting area of research.

Now, let us recall some basic concepts on complex dynamics. Given a rational function $R: \hat{\mathbb{C}} \to \hat{\mathbb{C}}$, where $\hat{\mathbb{C}}$ is the Riemann sphere, the *orbit of a point* $z_0 \in \hat{\mathbb{C}}$ is defined as $\left\{z_0, R(z_0), R^2(z_0), ..., R^n(z_0), ...\right\}$. A point $z^* \in \hat{\mathbb{C}}$ is called a *fixed point* of $R(z)$ if it verifies that $R(z^*) = z^*$. Moreover, z^* is called a *periodic point* of period $p > 1$ if it is a point such that $R^p(z^*) = z^*$ but $R^k(z^*) \neq z_0$, for each $k < p$. Moreover, a point z^* is called *pre-periodic* if it is not periodic but there exists a $k > 0$ such that $R^k(z^*)$ is periodic.

There exist different types of fixed points depending on their associated multiplier $|R'(z)|$. Taking the associated multiplier into account, a fixed point z^* is called: *superattracting* if $|R'(z^*)| = 0$, *attracting* if $|R'(z^*)| < 1$, *repulsive* if $|R'(z^*)| > 1$, and *parabolic* if $|R'(z^*)| = 1$. The fixed

http://dx.doi.org/10.5772/64106

point operator of any iterative method on an arbitrary polynomial $p(z)$ is a rational function. The fixed points of this rational function that do not correspond to the roots of the polynomial $p(z)$ are called *strange fixed points*. On the other hand, a *critical point* z^* is a point satisfying $|R'(z^*)| = 0$.

The basin of attraction of an attractor α is defined as $\mathcal{A}(\alpha) = \{z_0 \in \hat{\mathbb{C}};\ R^n(z_0) \to \alpha, n \to \infty\}$. The Fatou set of the rational function $\boldsymbol{R}$, $\mathcal{F}(R)$ is the set of points $z \in \hat{\mathbb{C}}$ whose orbits tend to an attractor (fixed point, periodic orbit or infinity). Its complement in $\mathbb{C}$ is the Julia set, $\mathcal{J}(R)$. That means that the basin of attraction of any fixed point belongs to the Fatou set and the boundaries of these basins of attraction belong to the Julia set.

Some other concepts are a key fact in this kind of analysis, as the immediate basin of attraction of an attracting fixed point α (considered as a periodic point of period 1), as the connected component of the basin containing α. This concept is directly related with the existence of critical points, as can be seen in the following classical result, due to Fatou and Julia.

Theorem 3.1. *Let R be a rational function. The immediate basins of attraction of any attracting periodic point hold, at least, a critical point.*

If a scaling theorem can be established, the qualitative behavior of the class of iterative schemes is analyzed on a generic quadratic polynomial $p(z) = z^2 - c$, its dynamics being analytically conjugated by affine transformations. Then, the fixed points of their associated rational function (being or not roots of the polynomial) are calculated and it is studied if they are stable (that is, if the successive iterations of the method converge to them) or if they are unstable, repelling the iterations near them. Finally, the calculation of critical points and their use as starting points of the iterative process will allow us, by applying Theorem 3.1, to find all the values of the parameter (that is, methods of the family) that do not converge to the roots of the polynomial, resulting in unstable behavior (attracting periodic orbits, chaos, etc.).

This kind of analysis has been developed by Cordero et al. [46] on the fourth-order biparametric Ostrowski-Chun family, whose iterative expression was shown in Eq. (5). In this manuscript, the strange fixed points and free independent critical points of the fixed point rational operator

$$O_p(z, a_1, b_2) = -z^4 \frac{(z+1)^2(z^2+4z+5) - a_1(b_2^2 - b_2(z^3+4z^2+5z+4) + 2(z+1)^2(z+2))}{z^4(a_1(b_2-2)^2-5) + z^3(-5a_1(b_2-2)-14) - 2z^2(2a_1(b_2-2)+7) - z(a_1(b_2-2)+6) - 1}$$

have been identified. As the behavior of the elements of the family depends on two parameters, only real values have been considered in the plane (a_1, b_2) in order to calculate the multipliers of the strange fixed points: $z = 1$ and

$$ex_1 = \frac{1}{12}(A - \sqrt{3B} - \sqrt{6(C+D)}),$$
$$ex_2 = \frac{1}{12}(A - \sqrt{3B} + \sqrt{6(C+D)}),$$
$$ex_3 = \frac{1}{12}(A + \sqrt{3B} - \sqrt{6(C+D)}),$$
$$ex_4 = \frac{1}{12}(A + \sqrt{3B} + \sqrt{6(C+D)}),$$

where A, B, C, and D depend on both parameters a_1 and b_2. In **Figure 1**, the stability function of strange fixed point $z = 1$ is represented, $\left|O'_p(1, a_1, b_2)\right|$. The orange region corresponds to real values of parameters a_1 and b_2 where $z = 1$ is attracting, meanwhile gray region includes values of a_1 and b_2 where this strange fixed point is repulsive, and therefore, the corresponding iterative methods have better behavior. Similar studies can be made on the rest of strange fixed points, searching for a more stable position of plane (a_1, b_2).

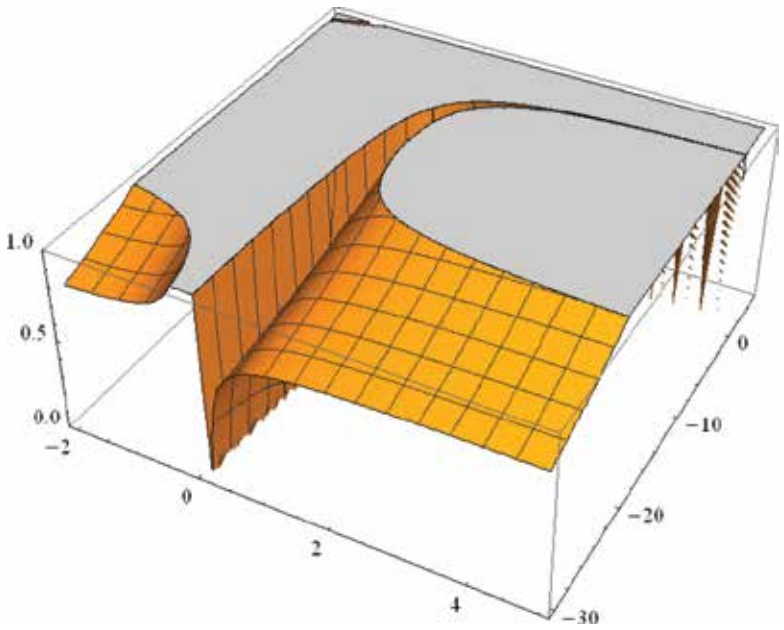

Figure 1. Stability function of $z = 1$ as fixed point of operator $O_p(z, a_1, b_2)$.

On the other hand, as the dynamics of critical points could lead to a Fatou component, their associated parameter planes are represented to see the behavior of the method when the initial estimate is a critical point. In this case, $z = -1$ and $cr_i, i = 1, 2, 3, 4$, are free critical points of $O_p(z, \boldsymbol{a_1}, \boldsymbol{b_2})$ such that $cr_1 = \frac{1}{cr_2}$ and $cr_3 = \frac{1}{cr_4}$, that will be used to calculate the different

http://dx.doi.org/10.5772/64106

parameter planes associated with the fixed point operator. As $z = -1$ is a pre-image of $z = 1$, only two critical points can be considered as free and independent: cr_2 and cr_4.

The parameter space of a free critical point is obtained by associating each point of the parameter plane with real values of free parameters a_1 and b_2, so every point on the plane represents a different member of the iterative family. These parameter planes (one of them can be seen in **Figure 2**) have been created using a vectorized version of the MATLAB code programs presented by Chicharro et al. [47], with 800 × 800 different combinations of $\boldsymbol{a_1}$ and $\boldsymbol{b_2}$. Black points correspond to the parameter values for which the associated iterative method does not converge to the conjugated values of the zeros of polynomial $p(z)$ with a tolerance of 10^{-3} after 500 iterations, taking the same free independent critical point as the initial estimate.

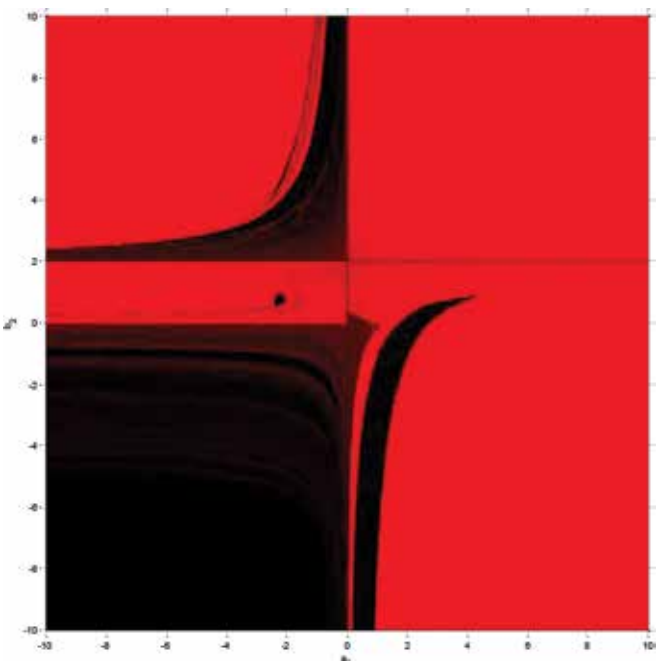

Figure 2. Parameter plane of operator $O_p(z, a_1, b_2)$.

Points shown in red in **Figure 2** (and also simultaneously red in the rest of parameter planes corresponding to other free independent critical points of the rational function) correspond to the most stable methods of the family. In these terms, **Figure 3** shows the dynamical plane associated with one of these stable elements of the family: each point with 800 × 800 mesh in the complex plane is an initial estimation for the iterative method. This point has an associated color depending on the convergence of the method after 80 iterations: if it converges to a fixed point that corresponds to a root of $p(z)$, then a color (orange and blue) is assigned (see **Figure 3a**). If not, then it diverges or converges to any other element (attracting strange fixed point, periodic orbit, etc.), and it is painted in black (see **Figure 3b**). In the former case, the orbit of an initial point in a black region is marked in yellow.

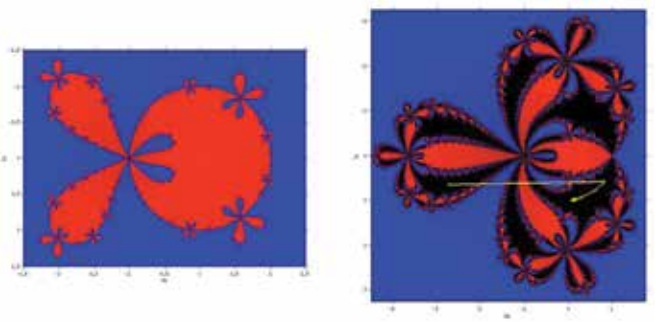

Figure 3. Dynamical plane of operator $O_p(z, a_1, b_2)$.

Selected values of parameters $\boldsymbol{a_1}$ and $\boldsymbol{b_2}$ in these red regions were tested numerically, not only on scalar equations, but also on multidimensional problems. These tests showed that good performance of the member of the family observed in the dynamical study could also be noticed in the multidimensional case.

However, very recently real multivariate dynamical tools have revealed also to be a reliable implement to analyze the stability of iterative methods, specially designed for solving nonlinear systems. In order to study the stability of the fixed points of vectorial rational functions associated with iterative schemes for solving nonlinear systems, a new tool was presented by Cordero et al. [33]. In it, the authors checked the consistence of this tool by applying it on known methods as Newton's and Traub's schemes and also on a family of parametric iterative procedures

$$
\begin{aligned}
y^{(k)} &= x^{(k)} - \theta[F'(x^{(k)})]^{-1}F(x^{(k)}),\\
z^{(k)} &= x^{(k)} - [F'(x^{(k)})]^{-1}(\theta F(x^{(k)}) + F(y^{(k)})),\\
x^{(k+1)} &= x^{(k)} - [F'(x^{(k)})]^{-1}(\theta F(x^{(k)}) + F(y^{(k)}) + F(z^{(k)})),
\end{aligned}
$$

Presented by Hueso et al. [48] and denoted by HMT.

In order to analyze, under a multidimensional point of view, the qualitative behavior of this family, the Cordero et al. [33] introduced some concepts of multidimensional real discrete dynamics that we will recall in the following; some of them are the natural extensions of the defined concepts in complex dynamics, but others will be defined, as being different from those.

Let us denote by $G(x)$ the vectorial fixed-point function associated with the iterative method or family on polynomial $p(x)$. The dynamical behavior of the orbit of a point of $\mathbb{R}^n$ can be classified depending on its asymptotic behavior. In this way, a point x^* is a fixed point of G if $G(x^*) = x^*$.

http://dx.doi.org/10.5772/64106

Theorem 3.2. [43, page 558] *Let* $G:\mathbb{R}^n \to \mathbb{R}^n$ be *a function in* $\mathcal{C}^2$. *Assume* x^* *is a period-k point. Let* $\lambda_1, \lambda_2, ..., \lambda_n$ *be the eigenvalues of* $G'(x^*)$.

a) *If all the eigenvalues* λ_j *have* $|\lambda_j| < 1$, *then* x^* *is attracting.*

b) *If one eigenvalue* λ_{j0} *has* $|\lambda_{j_0}| > 1$, *then* x^* *is unstable, that is, repelling or saddle.*

c) *If all the eigenvalues* λ_j *have* $|\lambda_j| > 1$, *then* x^* *is repelling.*

Moreover, a fixed point is called *hyperbolic* if all the eigenvalues λ_j of $G'(x^*)$ have $|\lambda_j| \neq 1$. Indeed, if there is an eigenvalue λ_i such that $|\lambda_i| < 1$ and also there exists an eigenvalue λ_j such that $|\lambda_j| > 1$, the hyperbolic point is called *saddle point*.

To avoid the calculation of spectrum of $G'(x^*)$, the authors proposed by Cordero et al. [33] a result that, being consistent with the previous theorem, provided a practical tool to classify the stability of fixed points in many multidimensional cases.

Proposition 3.1. *Let* x^* *be a fixed point of G. then,*

a) *if* $\left|\frac{\partial g_i(x^*)}{\partial x_j}\right| < \frac{1}{n}$, *for all* $\{1, 2, ..., n\}$, *then* x^* *is attracting.*

b) *if* $\left|\frac{\partial g_i(x^*)}{\partial x_j}\right| = 0$, *for all* $i, j \epsilon \{1, 2, ..., n\}$, *then* x^* *is superattracting.*

c) *if* $\left|\frac{\partial g_i(x^*)}{\partial x_j}\right| > \frac{1}{n}$, *for all* $j \epsilon \{1, 2, ..., n\}$, *then* x^* *is unstable and lies at the Julia set.*

being the coordinate functions of the fixed point multivariate function G.

Let us remark that if the order of convergence of the iterative method is at least two, then the roots of the nonlinear function are superattracting fixed points of the vectorial rational function corresponding to the iterative scheme. If a fixed point is not a root of the nonlinear function, it is called strange fixed point and its character can be analyzed in the same manner.

The concept of critical point can be defined following the idea of multivariate convergence of iterative methods.

Definition 3.1. *A fixed point* x^* *is a critical point of G if its coordinate functions* $g_i(x)$ *satisfy* $\frac{\partial g_i(x^*)}{\partial x_j} = 0$, *for all* $i, j \epsilon \{1, 2, ..., n\}$.

From this definition, a superattracting fixed point will also be a critical point of the operator and, from numerical point of view, the iterative scheme will have, at least, quadratic order of convergence. A critical point that is not root of $p(x)$ will be called free critical point.

The stability of this family is studied on two systems of real quadratic polynomials, $p(x) = 0$ and $q(x) = 0$, where

$$p(x_1,x_2)=\begin{cases}x_1^2-1,\\ x_2^2-1,\end{cases} and \quad q(x_1,x_2)=\begin{cases}x_1^2-1,\\ x_2^2-1.\end{cases}$$

The components of the iteration function G are, in case of $p(x)$,

$$g_1(x_1,x_2)=-\frac{8\theta^2x_1^2(x_1^2-1)^3+\theta^4(x_1^2-1)^4-16x_1^4(3x_1^4+6x_1^2-1)}{128x_1^7},$$

$$g_2(x_1,x_2)=-\frac{8\theta^2x_2^2(x_2^2-1)^3+\theta^4(x_2^2-1)^4-16x_2^4(3x_2^4+6x_2^2-1)}{128x_2^7}.$$

It was proved by Cordero et al. [33] that the number of fixed points of the vectorial rational function $G(x)$ associated with HMT iterative method on $p(x)$ is 64, four of them (those corresponding to the roots of p(x)) are superattractive for any value of θ. Moreover, there are 48 unstable strange fixed points and the character of the final 12 real strange fixed points is simultaneously attractive for two ranges of values of θ, [*−0.3847551*, *−0.3838109*] and [*0.3838109*, *0.3847551*], being superattractive if $\theta \approx -0.3838109$ or $\theta \approx 0.3838109$.

In **Figure 4**, we can see the dynamical plane of HMT method for $\theta \approx -0.3838109$, where those 12 attracting strange fixed points appear as white circles, with small basins of attraction. The roots of p(x) appear as white stars, in their own basins of attraction.

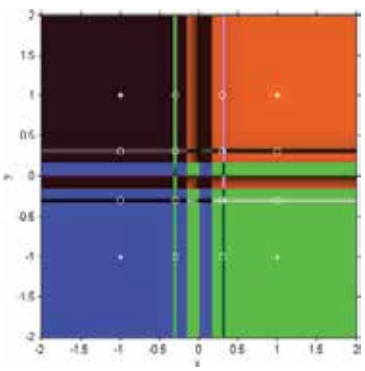

Figure 4. Dynamical plane of multivariate HMT method on $p(x)$ for $\theta \approx -0.3838109$.

The parametric plots, as the extension of parameter planes in multivariable case, were revealed to be an interesting procedure that allowed to detect the most stable and unstable elements of a family of iterative methods. By using the information that gives us the iteration of the elements of the family on the different free critical points, we can know the global behavior of the family depending on the value of the parameter. Orbits of each free critical point are

http://dx.doi.org/10.5772/64106

showed in **Figure 5**. In each one of these pictures, a different free critical point is used as initial guess of each member of the class of iterative schemes, taking values of parameter θ in [−5,5]. The values of θ corresponding to the members of the family are placed in the abscissa axis and the ordinate axis corresponds to 0.1, 0.2, 0.3, or 0.4 if the iterative process has converged to each one of the solutions of the quadratic polynomial real system, $p(x)$, respectively. Moreover, the ordinate of a point is −0.1 if the process diverges and it is null in other cases.

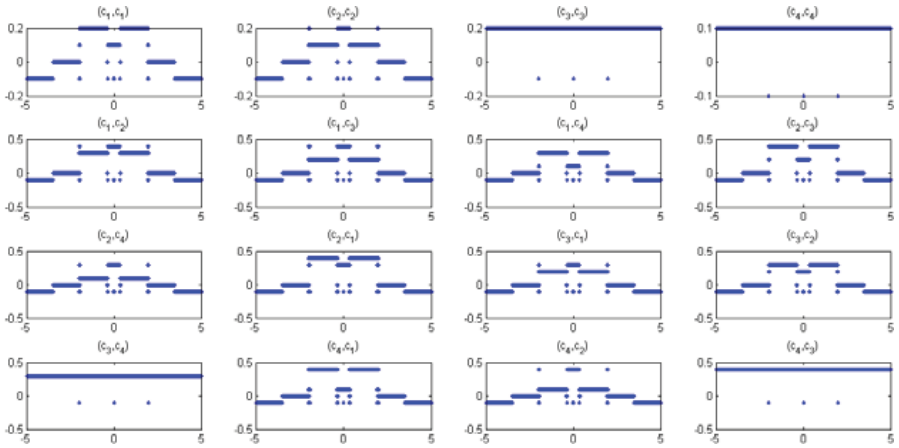

Figure 5. Different parameter plots of HMT family on $p(x)$.

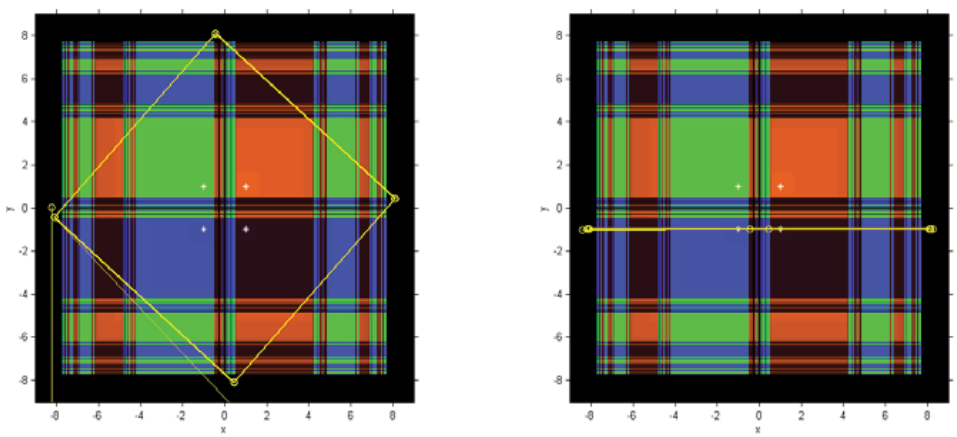

Figure 6. 4-Periodic orbits in HMT family.

The goal of these graphics is to show the elements of the family HMT (that is, the values of parameter θ) that present unstable behavior (attracting strange fixed points, periodic orbits,

…). These plots were obtained by using 20,000 subintervals, a maximum of 40 iterations and an error estimation of 10^{-3}, when iterates tend to a fixed point.

Thanks to this analysis, some non-desirable behavior was detected, as attracting periodic orbits of different periods, as can be observed in **Figure 6**, where two periodic orbits of period four are showed in yellow.

4. Numerical results

In this section, we are going to check the numerical performance of some elements of family (Eq. 10) compared with the multidimensional version of some other known methods of the same order and also with Newton's one.

Definition 4.1. *Let ξ be a zero of function F and suppose that $x^{(k-1)}$, $x^{(k)}$ and $x^{(k+1)}$ are three consecutive iterations close to ξ. Then, the computational order of convergence p can be approximated using the formula* (see Ref. [12]).

$$p \approx \frac{\ln(\| x^{(k+1)} - x^{(k)} \| / \| x^{(k)} - x^{(k-1)} \|)}{\ln(\| x^{(k)} - x^{(k-1)} \| / \| x^{(k-1)} - x^{(k-2)} \|)}$$

Numerical computations have been carried out in MATLAB, with variable precision arithmetic that uses floating point representation of 1000 decimal digits of mantissa. If $n > 1$, every iterate $x^{(k+1)}$ is obtained from the previous one, $x^{(k)}$, by adding one term of the form $A^{-1}b$. Matrix A and vector b are different according to the method used, but in any case the inverse calculation $-A^{-1}b$ is computed by solving the linear system $Ay = -b$, by using Gaussian elimination with partial pivoting. Nevertheless, if several systems with the same matrix A must be solved in iteration, LU factorization is made once and the corresponding triangular systems are solved by substitution. The stopping criterion used is $\left\| F\left(x^{(k)}\right) \right\| < 10^{-700}$ or $\left\| x^{(k+1)} - x^{(k)} \right\| < 10^{-700}$.

4.1. Molecular interaction problem

We are going to solve the equation of molecular interaction,

$$u_{xx} + u_{yy} = u^2, (x, y) \in [0,1] \times [0,1],$$

with the boundary conditions $u(x, 0) = 2x^2 - x + 1, u(x, 1) = 2, u(0, y) = 2y^2 - y + 1$ and $u(1, y) = 2$, for all (x, y) in their domain.

For approximating its solution, we are going to use central divided differences procedure to transform the problem in a nonlinear system of equations. This system is solved by means of the proposed methods of order four and another known ones.

http://dx.doi.org/10.5772/64106

The discretization process yields the nonlinear system of equations,

$$u_{i+1,j} - 4u_{i,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - h^2 u_{i,j}^2 = 0, i = 1,2,\ldots,nx, j = 1,2,\ldots,ny,$$

where $u_{i,j}$ denotes the estimation of the unknown $u(x_i,y_j)$, where $x_i = ih$ with $i = 1,2,\ldots, nx$ and $y_j = jk$ with $j = 1,2, \ldots, ny$ are the nodes in both variables, with $h = \frac{1}{nx}, k = \frac{1}{ny}$ and $nx = ny$.

For checking purposes, we consider $nx = ny = 4$, getting a mesh of 5 × 5 points. Applying the boundary conditions we have only nine unknowns, which we rename as:

$$\begin{aligned} x_1 &= u_{1,1}, \quad x_2 = u_{2,1}, \quad x_3 = u_{3,1}, \\ x_4 &= u_{1,2}, \quad x_5 = u_{2,2}, \quad x_6 = u_{3,2}, \\ x_7 &= u_{1,3}, \quad x_8 = u_{2,3}, \quad x_9 = u_{3,3}. \end{aligned}$$

Then, the nonlinear system associated with the partial differential equation, including the boundary conditions, can be expressed as

$$F(x) = Ax + \varphi(x) - b = 0,$$

where

$$A = \begin{pmatrix} M & -I & 0 \\ -I & M & -I \\ 0 & -I & M \end{pmatrix}, \quad M = \begin{pmatrix} 4 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 4 \end{pmatrix}$$

I being the 3 × 3 identity matrix, $\varphi(x) = h^2\left(x_1^2, x_2^2, ..., x_9^2\right)^T$ and $b = \left(\frac{7}{4}, 1, \frac{27}{8}, 1, 0, 2, \frac{27}{8}, 2, 4\right)^T$.

In this case,

$$F'(x) = A + 2h^2 diag(x_1, x_2, \ldots x_9).$$

For solving this system, we apply the extension for systems of Ostrowski's method (OM), Chun's scheme (CM), Jarratt's method (JM), Newton's method (NM), and three elements of family (Eq. 10) obtained by selecting stable values of the parameters.

$$\begin{aligned} MA &: a_1 = \frac{5}{4}, \; b_2 = 0, \\ MB &: a_1 = 1, \; b_2 = 1, \\ MC &: a_1 = 1, \; b_2 = 3. \end{aligned}$$

In **Table 4**, the approximated computational order of convergence ACOC, the number of iterations, the difference between the two last iterations and the residual of the function at the last iteration are shown, for each one of the methods. In all cases, the initial estimation is a null vector and the Euclidean norm is used in the calculation of the residuals.

Method	ACOC	Iteration	$\left\|x^{(k+1)} - x^{(k)}\right\|$	$\left\|F(x^{(k)})\right\|$
NM	1.9999	9	1.482e-413	6.448e-828
JM	3.9954	5	1.482e-413	1.976e-1007
OM	3.9964	5	1.482e-413	1.618e-1007
CM	3.9959	5	1.998e-353	1.618e-1007
MA	4.0519	5	5.362e-510	1.707e-2007
MB	3.9960	5	7.123e-362	1.409e-1449
MC	3.9960	5	3.110e-362	3.811e-1451

Table 4. Numerical results for molecular interaction problem.

All the checked schemes provide the same solution of the nonlinear system. In **Table 4**, we can observe that all the fourth-order methods have a similar performance, but we note that the lowest error corresponds to method MA, duplicating the number of exact digits with respect to the other ones.

5. Conclusions

Many problems in science and engineering are modeled in such a way that, for their solution, it is necessary to solve systems of nonlinear equations. Therefore, designing iterative methods for solving these types of problems is an important task and it is a fruitful area of research. In this chapter, a review of the different techniques for constructing iterative methods is presented. Moreover, it is shown that real discrete dynamics tools are useful for analyzing the stability of the designed methods, selecting those with good dynamical behavior. In the numerical section, a chemical problem is used for testing the presented methods and the theoretical results are confirmed.

Acknowledgements

This research was partially supported by Ministerio de Economía y Competitividad of Spain, MTM2014-52016-C2-2-P, and by Ministry of Higher Education Science and Technology of Dominican Republic, FONDOCYT 2014-1C1-088.

Author details

Alicia Cordero[1], Juan R. Torregrosa[1] and Maria P. Vassileva[2*]

*Address all correspondence to: maria.penkova@intec.edu.do

1 Multidisciplinary Mathematical Institute, Polytechnic University of Valencia, Spain

2 Technological Institute of Santo Domingo (INTEC), Santo Domingo, Dominican Republic

References

[1] Bruns D.D., Bailey J.E. Nonlinear feedback control for operating a nonisothermal CSTR near and unstable steady state. Chemical Engineering Science. 1977;32:257–264.

[2] Ezquerro J.A., Gutiérrez J.M., Hernández M.A., Salanova M.A. Chebyshev-like methods and quadratic equations. Revue d'analyse Numérique et de Théorie de l'approximation. 2000;28:23–35.

[3] He Y., Ding C. Using accurate arithmetics to improve numerical reproducibility and stability in parallel applications. Journal of Supercomputing. 2001;18:259–277.

[4] Iliev A., Kyurkchev N. Nontrivial methods in numerical analysis: select topics in numerical analysis. Saarbcken, Germany: Lap Lambert Academic Publishing; 2010.

[5] Zhang Y., Huang P. High-precision time-interval measurement techniques and methods. Progress in Astronomy. 2006;24(1):1–15.

[6] Abad M., Cordero A., Torregrosa J.R. A family of seventh-order schemes for solving nonlinear systems. Bulletin Mathématique Societe des Sciences Mathematiques de Roumanie. 2014;57(105):133–145.

[7] Artidiello S., Cordero A., Torregrosa J.R., Vassileva M.P. Design of high-order iterative methods for nonlinear systems by using weight-function procedure. Abstract and Applied Analysis. 2015;2015:12. DOI: 10.1155/2015/289029

[8] Babajee D.K.R., Dauhoo M.Z., Darvishi M.T., Karami A., Barati A. Analysis if two Chebyshev-like three-order methods free from second derivative for solving systems of nonlinear equation. Journal of Computational and Applied Mathematics. 2010;233(8):2002–2012.

[9] Traub J.F. Iterative methods for the solution of equations. New York: Prentice Hall; 1964.

[10] Ostrowski A.M. Solutions of equations and systems of equations. Academic Press; 1966.

[11] Cordero A., Torregrosa J.R. Variants of Newton's method for functions of several variables. Applied Mathematics and Computation. 2006;183:199–208.

[12] Cordero A., Torregrosa J.R. Variants of Newton's method using fifth-order quadrature formulas. Applied Mathematics and Computation. 2007;190:686–698.

[13] Cordero A., Torregrosa J.R. On interpolation variants of Newton's method for functions of several variables. Journal of Computational and Applied Mathematics. 2010;234:34–43.

[14] Frontini M., Sormani E. Third-order methods from quadrature formulae for solving systems of nonlinear equations. Applied Mathematics and Computation. 2004;149:771–782.

[15] Gerlach J. Accelerated convergence in Newton's method. SIAM Review. 1994;36(2):272–276.

[16] Ozban A.Y. Some new variants of Newton's method. Applied Mathematics Letters. 2014;17:677–682.

[17] Wang X., Zhang T., Qian W., Teng M. Seventh-order derivative-free iterative method for solving nonlinear systems. Numerical Algorithms. 2015;70:545–558.

[18] Weerakoon S., Fernando T.G.Y. A variant of Newton's method with accelerated third-order convergence. Applied Mathematics Letters. 2000;13:87–93.

[19] Cordero A., Torregrosa J.R., Vassileva M.P. Pseudo-composition: a technique to design predictor-corrector methods for systems of nonlinear equations. Applied Mathematics and Computation. 2012;218:11496–11508.

[20] Vassileva M.P. Métodos iterativos eficientes para la resolución de sistemas no lineales [thesis]. Valencia: Universitat Politècnica de València; 2011. 217 p. Available from: https://riunet.upv.es/bitstream/handle/10251/12892/tesisUPV3676.pdf?sequence=1&isAllowed=y

[21] Adomian G. Solving frontier problems of Physics: the decomposition method. The Netherlands: Kluwer Academic ed; 1994.

[22] Babajee D.K.R., Dauhoo M.Z., Darvishi M.T., Barati A. A note on the local convergence of iterative methods based on Adomian decomposition method and three-node quadrature root. Applied Mathematics and Computation. 2008;200(1):452–458.

[23] Cordero A., Martínez E., Torregrosa J.R. Iterative methods of order four and five for systems of nonlinear equations. Journal of Computational and Applied Mathematics. 2009;231(2):541–551.

[24] Liu Z., Zheng Q., Zhao P. A variant of Steffensen's method of fourth-order of convergence and its applications. Applied Mathematics and Computation. 2010;216:1978–1983.

[25] Arroyo V., Cordero A., Torregrosa J.R. Approximation of artificial satellites preliminary orbits: the efficiency challenge. Journal of Mathematical and Computer Modelling. 2011;54(7–8):1802–1807.

[26] Cordero A., Hueso J.L., Martínez E., Torregrosa J.R. A modified Newton-Jarratt's composition. Numerical Algorithms. 2010;55:87–99.

[27] Arroyo V., Cordero A., Torregrosa J.R., Vassileva M.P. Artificial satellites preliminary orbit determination by modified high-order Gauss methods. International Journal of Computer Mathematics. 2012;89(3):347–356.

[28] Sharma J.R., Guha R.K., Sharma R. An efficient fourth-order weighted-Newton method for systems of nonlinear equations. Numerical Algorithms. 2013;62:307–323.

[29] Artidiello S., Cordero A., Torregrosa J.R., Vassileva M.P. Multidimensional generalization of iterative methods for solving nonlinear problems by means of weight-functions procedure. Applied Mathematics and Computation. 2015;268:1064–1071.

[30] Cordero A., García-Maimó J., Torregrosa J.R., Vassileva M.P. Solving nonlinear problems by Ostrowski-Chun type parametric families. Journal of Mathematical Chemistry. 2015;53:430–449.

[31] Cordero A., Feng L., Magreñán A.A., Torregrosa J.R. A new fourth-order family for solving nonlinear problems and its dynamics. Journal of Mathematical Chemistry. 2015;53:893–910.

[32] Cordero A., Torregrosa J.R., Vassileva M.P. Increasing the order of convergence of iterative schemes for solving nonlinear systems. Journal of Computational and Applied Mathematics. 2013;252:86–94.

[33] Cordero A., Soleymani F., Torregrosa J.R. Dynamical analysis of iterative methods for nonlinear systems or how to deal with the dimension? Applied Mathematics and Computation. 2014;244:398–412.

[34] Kung H.T., Traub F.F. Optimal order of one-point and multipoint iteration. Journal ACM. 1974;21:643–651.

[35] Cordero A., Torregrosa J.R., Vassileva M.P. Weighted-Gaussian correction of Newton-type methods for solving nonlinear systems. Bulletin Mathematique de la Societe des Sciences Mathematiques de Roumanie. 2016;59(107):23–38.

[36] Chun C. Construction of Newton-like iterative methods for solving nonlinear equations. Numerical Mathematics. 2006;104:297–315.

[37] Ortega J.M., Rheinboldt W.C. Iterative solutions of nonlinear equations in several variables. Academic Press Inc.; 1970.

[38] Amat S., Busquier S., Bermúdez C., Plaza S. On two families of high order Newton type methods. Applied Mathematic Letters. 2012;25:2209–2217.

[39] Amat S., Busquier S., Plaza S. Chaotic dynamics of a third-order Newton-type method. Journal of Computational and Applied Mathematics. 2010;366:24–32.

[40] Babajee D.K.R., Cordero A., Soleymani F., Torregrosa J.R. On improved three-step schemes with high efficiency index and their dynamics. Numerical Algorithms. 2014;65(1):153–169.

[41] Campos B., Cordero A., Magreñán A.A., Torregrosa J.R., Vindel P. Study of a biparametric family of iterative methods. Abstract and Applied Analysis. 2014;2014:12.

[42] Chun C., Neta B., Kozdon J., Scott M. Choosing weight functions in iterative methods for simple roots. Applied Mathematics and Computation. 2014;227:788–800.

[43] Cordero A., Maimó J.G., Torregrosa J.R., Vassileva M.P., Vindel P. Chaos in King's iterative family. Applied Mathematics Letters. 2013;26:842–848.

[44] Cordero A., Torregrosa J.R., Vindel P. Dynamics of a family of Chebyshev-Halley type methods. Applied Mathematics and Computation. 2013;219:8568–8583.

[45] Magreñán A.A. Different anomalies in a Jarratt family of iterative root-finding methods. Applied Mathematics and Computation. 2014;233:29–38.

[46] Cordero A., Maimó J.G., Torregrosa J.R., Vassileva M.P. Stability of a fourth order biparametric family of iterative methods. Journal of Computational and Applied Mathematics. DOI: 10.1016/j.cam.2016.01.013

[47] Chicharro F.I., Cordero A., Torregrosa J.R. Drawing dynamical and parameter planes of iterative families and methods. The Scientific World Journal. 2013;2013:11.

[48] Hueso J.L., Martínes E., Torregrosa J.R. New modifications of Potra-Pták's method with optimal fourth and eighth orders of convergence. Journal of Computational and Applied Mathematics. 2010;234:2969–2976.

[49] Robinson R.C. An introduction to dynamical systems, continuous and discrete. Providence: American Mathematical Society; 2012.

Chapter 6

Nonlinear State and Parameter Estimation Using Iterated Sigma Point Kalman Filter: Comparative Studies

Marwa Chaabane, Imen Baklouti, Majdi Mansouri, Nouha Jaoua, Hazem Nounou, Mohamed Nounou, Ahmed Ben Hamida and Marie-France Destain

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/63728

Abstract

In this chapter, iterated sigma-point Kalman filter (ISPKF) methods are used for nonlinear state variable and model parameter estimation. Different conventional state estimation methods, namely the unscented Kalman filter (UKF), the central difference Kalman filter (CDKF), the square-root unscented Kalman filter (SRUKF), the square-root central difference Kalman filter (SRCDKF), the iterated unscented Kalman filter (IUKF), the iterated central difference Kalman filter (ICDKF), the iterated square-root unscented Kalman filter (ISRUKF) and the iterated square-root central difference Kalman filter (ISRCDKF) are evaluated through a simulation example with two comparative studies in terms of state accuracies, estimation errors and convergence. The state variables are estimated in the first comparative study, from noisy measurements with the several estimation methods. Then, in the next comparative study, both of states and parameters are estimated, and are compared by calculating the estimation root mean square error (RMSE) with the noise-free data. The impacts of the practical challenges (measurement noise and number of estimated states/parameters) on the performances of the estimation techniques are investigated. The results of both comparative studies reveal that the ISRCDKF method provides better estimation accuracy than the IUKF, ICDKF and ISRUKF. Also the previous methods provide better accuracy than the UKF, CDKF, SRUKF and SRCDKF techniques. The ISRCDKF method provides accuracy over the other different estimation techniques; by iterating maximum a posteriori estimate around the updated state, it re-linearizes the measurement equation instead of depending on the predicted state. The results also represent that estimating more parameters impacts the estimation accuracy as well as the convergence of the estimated parameters and states. The ISRCDKF provides improved state accuracies than the other techniques even with abrupt changes in estimated states.

Keywords: Kalman filter, sigma point, state estimation, parameter estimation, nonlinear system

1. Introduction

Dynamic state-space models [1–3] are useful for describing data in many different areas, such as engineering [4–8], biological data [9, 10], chemical data [11, 12], and environmental data [8, 13–15]. Estimation of the state and model parameters based on measurements from the observation process is an essential task when analyzing data by state-space models. Bayesian estimation filtering represents a solution of considerable importance for this type of problem definition as demonstrated by many existing algorithms based on the Bayesian filtering [16–25]. The Kalman filter (KF) [26–29] has been extensively utilized in several science applications, such as control, machine learning and neuroscience. The KF provides an optimum solution [28], when the model describing the system is supposed to be Gaussian and linear. However, the KF is limited when the model is considered to be nonlinear and present non-Gaussian modeling assumptions. In order to relax these assumptions, the extended Kalman filter (EKF) [26, 27, 30–32], the unscented Kalman filter (UKF) [33–36], the central difference Kalman filter (CDKF) [37, 38], the square-root unscented Kalman filter (SRUKF) [39, 40], the square-root central difference Kalman filter (SRCDKF) [41], the iterated unscented Kalman filter (IUKF) [42, 43], the iterated central difference Kalman filter (ICDKF) [44, 45], the iterated square-root unscented Kalman filter (ISRUKF) [46] and the iterated square-root central difference Kalman filter (ISRCDKF) [47] have been developed. The EKF [26] linearizes the model describing the system to approximate the covariance matrix of the state vector. However, the EKF is not always performing especially for highly nonlinear or complex models. On behalf of linearizing the model, a class of filters called the sigma-point Kalman filters (SPKFs) [48] uses a statistical linearization technique which linearizes a nonlinear function of a random variable via a linear regression. This regression is done between n points drawn from the prior distribution of the random variable, and the nonlinear functional evaluations of those points. The sigma-point family of filters has been proposed to address the issues of the EKF by making use of a deterministic sampling approach. In this approach, the state distribution is approximated and represented by a set of chosen weighted sample points which capture the true mean and covariance of the state vector. These points are propagated through the true nonlinear system and capture the posterior mean and the covariance matrix of the state vector accurately to the third order (Taylor series expansion) for any nonlinearity. As part of the SPKF family, the UKF [26, 27, 33] has been developed. It uses the unscented transformation, in which a set of samples (sigma points) are propagated and selected by the nonlinear model, providing more accurate approximations of the covariance matrix and mean of the state vector. However, the UKF technique has the limit of the number of sigma-points which are not so large and cannot represent complicated distributions. Another filter in the SPKF family is the central difference Kalman filter (CDKF) [37, 38]. It uses the Stirling polynomial interpolation formula. This filter has the benefit over the UKF in using only one parameter when generating the sigma-point.

http://dx.doi.org/10.5772/63728

To add some benefits of numerical stability, the SRUKF and the SRCDKF [41] have been developed. The advantage of these filters is that they ensured positive semidefiniteness of the state covariances. The iterated sigma-point Kalman filter (ISPKF) methods employ an iterative procedure within a single measurement update step by resampling the sigma-point till a termination criterion, based on the minimization of the maximum likelihood estimate, is satisfied.

The objectives of this chapter are threefold: (i) To estimate nonlinear state variables and model parameters using SPKF methods and extensions through a simulation example. (ii) To investigate the effects of practical challenges (such as measurement noise and number of estimated states/parameters) on the performances of the techniques. To study the effect of measurement noise on the estimation performances, several measurement noise levels will be considered. Then, the estimation performances of the techniques will be evaluated for different noise levels. Also, to study the effect of the number of estimated states/parameters on the estimation performances of all the techniques, the estimation performance will be studied for different numbers of estimated states and parameters. (iii) To apply the techniques to estimate the state variables as well as the model parameters of second-order LTI system. The performances of the estimation techniques will be compared to each other by computing the execution times as well as the estimation root mean square error (RMSE) with respect to the noise-free data.

2. State estimation problem

Next, we present the formulation of the state estimation problem.

2.1. Problem description and formulation

The state estimation problem for a system of nonlinear complex model is described as follows:

$$\begin{aligned} \dot{x} &= g(x, u, \theta, w) \\ y &= l(x, u, \theta, v) \end{aligned} \tag{1}$$

where $x \in \mathrm{R}^{\mathrm{n}}$ is the state variable vector, $y \in \mathrm{R}^{\mathrm{m}}$ is the measurement vector, $\theta \in \mathrm{R}^{\mathrm{q}}$ is the unknown vector, $u \in \mathrm{R}^{\mathrm{p}}$ is the input variable vector, $w \in \mathrm{R}^{\mathrm{n}}$ and $v \in \mathrm{R}^{\mathrm{m}}$ are respectively process and measurement noise vectors, and g and l are nonlinear differentiable functions. The discretization of the model (1) is presented as follows:

$$\begin{aligned} x_k &= \mathrm{f}(x_{k-1}, u_{k-1}, \theta_{k-1}, w_{k-1}) \\ y_k &= h(x_k, u_k, \theta_k, v_k) \end{aligned} \tag{2}$$

which describes the state variables at some time step (k) in terms of their values at a previous time step ($k-1$). Since we are interested to estimate the state vector $x_{k'}$ as well as the parameter vector $\theta_{k'}$ the parameter vector is assumed to be presented as follows:

$$\theta_k = \theta_{k-1} + \gamma_{k-1} \tag{3}$$

This means that it corresponds to a stationary process, with an identity transition matrix, driven by white noise. In order to include the parameter vector θ_k into the state estimation problem, let us define a new state vector z_k that augments the state vector x_k and the parameter vector θ_k as follows:

$$z_k = \begin{bmatrix} x_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} f(x_{k-1}, u_{k-1}, v_{k-1}, \theta_{k-1}) \\ \theta_{k-1} + \gamma_{k-1} \end{bmatrix} \tag{4}$$

where $z_k \in \mathrm{R}^{n+q}$. Also, defining the augmented noise vector as:

$$\varepsilon_{k-1} = \begin{bmatrix} v_{k-1} \\ \gamma_{k-1} \end{bmatrix} \tag{5}$$

The model (2) can be written as:

$$z_k = \mathcal{F}(z_{k-1}, u_{k-1}, \varepsilon_{k-1}) \tag{6}$$

$$y_k = \mathcal{R}(z_k, u_k, v_k) \tag{7}$$

where $\mathcal{F}$ and $\mathcal{R}$ are differentiable nonlinear functions. Thus, the objective here is to estimate the augmented state vector $z_{k'}$ given the measurement vector y_k.

3. Description of state estimation methods

3.1. UKF

The UKF is a SPKF that uses the unscented transformation. This transformation is a method for calculating the statistics of a random variable that undergoes a nonlinear mapping. It is built on the theory that "it is easier to approximate a probability distribution than an arbitrary nonlinear function".

The state distribution is represented by a Gaussian random variable (GRV) and by a set of deterministically chosen points. These points capture the true mean and covariance of the GRV

http://dx.doi.org/10.5772/63728

and also capture the posterior mean and covariance accurately to the second order for any nonlinearity and to the third order for Gaussian inputs. Suppose that GRV $z \in R^L$ characterized by a mean $\bar{z}$ and covariance P_z is used in the model. This variable is transformed by a nonlinear function $y = f(z)$. To reach the statistics of y, a $2L + 1$ sigma vector is defined as follows:

$$\begin{aligned} z_0 &= \bar{z} \\ z_i &= \bar{\bar{z}} + (\sqrt{(L+\lambda)P_z})_i \qquad i = 1,..,L \\ z_i &= \bar{\bar{z}} - (\sqrt{(L+\lambda)P_z})_i \qquad i = L+1,..,2L \end{aligned} \tag{8}$$

where L is the dimension of the state z, $\lambda = e^2(L + \kappa) - L$ is a scaling parameter and $(\sqrt{(L+\lambda)P_z})_i$ denotes the ith column of the matrix square root. The constant $10^{-4} < e < 1$ defines the spread of the sigma-points around $\bar{z}$. The constant κ is a scaling parameter which is usually set to zero or $3 - L$ [30].

Then, these sigma-points are propagated through the nonlinear function,

$$\mathrm{Y_i} = \mathrm{f(Z_i)} \quad \mathrm{i} = 0,..,2\mathrm{L} \tag{9}$$

And the mean and covariance matrix of y can be approximated as weighted sample mean and covariance of the transformed sigma-point of Y_i as follows:

$$\bar{y} = \sum_{i=0}^{2L} W_i^{(m)} Y \quad and \quad P_{\tilde{y}_k} = \sum_{i=0}^{2L} W_i^{(c)} (Y_i - \bar{y})(Y_i - \bar{y}) \tag{10}$$

where the weights are given by

$$\begin{aligned} W_0^{(m)} &= \frac{\lambda}{\lambda+L} \\ W_0^{(c)} &= \frac{\lambda}{\lambda+L} + (1 - e^2 + \zeta) \\ W_i^{(m)} &= W_i^{(c)} = \frac{1}{2(\lambda+L)} \qquad i = 1,..,2L \end{aligned} \tag{11}$$

The parameter ξ is used to integrate prior knowledge about the distribution of z.

The algorithm of the UKF includes two steps: prediction and update. In the prediction step, we calculate the predicted state estimate $\hat{z}_k^-$ and the predicted estimate covariance P_k^-. In the update step, we calculate the updated state estimate $\hat{z}_k$ and the updated estimate covariance P_k after calculating the innovation residual $P_{z_k y_k}$ and the optimal Kalman gain K_k.

The UKF technique is summarized in Algorithm 1.

3.2. CDKF method

The CDKF is another filter from the family of SPKF. This filter is based on Sterling polynomial interpolation formula instead of the unscented transformation used in UKF. The CDKF is similar to the UKF with the same or superior performance. However, it has an advantage over the UKF that it uses only one parameter instead of three parameters in the UKF. The CDKF uses a symmetric set of $(2L + 1)$ sigma-point which are calculated as follows,

$$\begin{aligned} z_0 &= \hat{z} \\ z_i &= \hat{z} + (h\sqrt{P_z})_i \qquad i = 1,..,L \\ z_i &= \hat{z} - (h\sqrt{P_z})_i \quad i = L+1,..,2L \end{aligned} \tag{12}$$

where L is the dimension of the state z, h is a scaling parameter (the optimal value is $h = \sqrt{3}$) and i indicates the ith column of the matrix.

These sigma-points are propagated through the nonlinear function to form the set of the posterior sigma-point,

$$Y_i = f(\Psi_i) \quad i = 0,..,2L \tag{13}$$

Within the above results, the sterling approximation estimates of the mean $\hat{z}$, covariance P_y and cross covariance $P_{z,y}$ are obtained through a linear regression of weighted point,

$$\hat{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} Y_i \tag{14}$$

$$P_{\bar{y}_k} = \sum_{i=1}^{L} \left[W_i^{(c_1)} (Y_{i,k|k-1} + Y_{i+L,k|k-1})^2 + W_i^{(c_2)} (Y_{i,k|k-1} + Y_{i+L,k|k-1} + 2Y_0)^2 \right] \tag{15}$$

$$P_{z_k y_k} = \sqrt{W_1^{(c_1)} P_z} \left[Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1} \right]^T \tag{16}$$

The set of corresponding weights for the mean $W_i^{(m)}$ which are used to compute the posterior mean is defined as:

$$W_0^{(m)} = \frac{h^2 - L}{h^2}, W_i^{(m)} = \frac{1}{2h^2} \tag{17}$$

And the set of corresponding weights for the covariance $W_0^{(c)}$ which is used to recover the covariance and the cross-covariance is defined as,

http://dx.doi.org/10.5772/63728

$$W_i^{(c1)} = \frac{1}{4h^2}, \quad W_i^{(c2)} = \frac{h^2-1}{4h^4} \quad \text{i} = 1,..,2\text{L} \tag{18}$$

The CDKF technique is summarized in Algorithm 2.

3.3. SRUKF method

One drawback of the UKF is that it requires the calculation of the matrix square-root $S_k S_k^T = P_{k'}$ at each time step. That is why a square-root form of the UKF has been developed to reduce the computational complexity. In this new method the covariance matrix S_k will be propagated directly, avoiding to refactorize at each time step [34].

The SRUKF is initialized as follows:

$$\hat{z}_0 = E[z_0] \; And \; S_0 = chol\{E[(z_0 - \hat{z}_0)(z_0 - \hat{z}_0)']\} \tag{19}$$

$$\Psi_{k-1} = [\hat{z}_{k-1} \quad \hat{z}_{k-1} + hS_{k-1} \quad \hat{z}_{k-1} - hS_{k-1}] \tag{20}$$

Algorithm 1: UKF algorithm

- *Initialization step:*

$$z_0 = E[z_0] \; and \; P_{z_0} = [(z - \hat{z}_0)(z - \hat{z}_0)^T]$$

- *Prediction step:*

$$\Psi_{k-1} = \left[\hat{z}_{k-1} \quad \hat{z}_{k-1} + \sqrt{(L+\lambda)P_z} \quad \hat{z}_{k-1} - \sqrt{(L+\lambda)P_z}\right]$$

$$\Psi_{k|k-1} = f(\Psi_{k-1})$$

$$\hat{z}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \; \Psi_{i,k|k-1}$$

$$P_k^- = \sum_{i=0}^{2L} W_i^{(c)} \; (\Psi_{i,k|k-1} - \hat{z}_k^-)(\Psi_{i,k|k-1} - \hat{z}_k^-)^T$$

$$Y_{k|k-1} = h\left[\Psi_{k|k-1}\right]$$

$$\hat{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,k|k-1}$$

- *Estimation (update) step:*

$$p_{\tilde{y}_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[Y_{i,k|k-1} - \hat{y}_k^-\right]\left[Y_{i,k|k-1} - \hat{y}_k^-\right]^T$$

$$P_{z_k y_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[\Psi_{i,k|k-1} - \hat{z}_k^-\right]\left[Y_{i,k|k-1} - \hat{y}_k^-\right]^T$$

$$K_k = P_{z_k y_k} P_{\tilde{y}_k}^{-1}$$

$$\hat{z}_k = \hat{z}_k^- + K_k(y_k - \hat{y}_k^-)$$

$$P_k = P_k^- - K_k P_{\tilde{y}_k} K_k^T$$

Return the augmented state estimation $\hat{z}_k$

Algorithm 2: CDKF algorithm

- *Initialization step:*

$$z_0 = E\left[z_0\right] and\ P_{z_0} = \left[(z - \hat{z}_0)(z - \hat{z}_0)^T\right]$$

- *Prediction step:*

$$\Psi_{k-1} = \left[\hat{z}_{k-1} \quad \hat{z}_{k-1} + h\sqrt{P_z} \quad \hat{z}_{k-1} - h\sqrt{P_z}\right]$$

$$\Psi_{k|k-1} = f(\Psi_{k-1})$$

$$\hat{z}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1}$$

http://dx.doi.org/10.5772/63728

$$P_k^- = \sum_{i=0}^{L}\left[W_i^{(c_1)}(\Psi_{i,k|k-1} - \Psi_{L+i,k|k-1})^2 + W_i^{(c_2)}(\Psi_{i,k|k-1} + \Psi_{L+i,k|k-1} - 2\Psi_{0,k|k-1})^2\right]$$

$$Y_{k|k-1} = h\left[\Psi_{k|k-1}\right]$$

$$\hat{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,k|k-1}$$

- *Estimation (update) step:*

$$P_{\bar{y}_k} = \sum_{i=0}^{L}\left[W_i^{(c_1)}(Y_{i,k|k-1} - Y_{L+i,k|k-1})^2 + W_i^{(c_2)}(Y_{i,k|k-1} + Y_{L+i,k|k-1} - 2Y_0)^2\right]$$

$$P_{z_k y_k} = \sqrt{W_1^{(c_1)} P_k^-}\left[Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1}\right]^T$$

$$K_k = P_{z_k y_k} P_{\bar{y}_k}^{-1}$$

$$\hat{z}_k = \hat{z}_k^- + K_k(y_k - \hat{y}_k^-)$$

$$P_k = P_k^- - K_k P_{\bar{y}_k} K_k^T$$

Return the augmented state estimation $\hat{z}_k$

The Cholesky factorization decomposes a symmetric, positive-definite matrix into the product of a lower triangular matrix and its transpose. This new matrix is utilized directly to obtain the sigma-point: The scaling constant h is expressed as $h = \sqrt{L\alpha^2}$, where α is a tunable parameter less than one.

In order to predict the current attitude based on each sigma-point, these sigma-points are transformed through the nonlinear process system

$$\Psi_{k/k-1} = f[\Psi_{k-1}] \tag{21}$$

Then, the state mean and the square-root covariance are estimated and calculated through the transformed sigma-point as follows:

$$\hat{z}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1} \tag{22}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c)}}\left(\Psi_{1:2L,k/k-1} - \hat{Z}_k^-\right)\sqrt{R^w}\right]\right\} \tag{23}$$

$$S_k^- = cholupdate\left\{S_k^-,\ \Psi_{0,k} - \hat{Z}_k^-,\ W_0^{(c)}\right\} \tag{24}$$

where $W_0^{(c)} = 2\left(1 - \alpha^2 + \frac{1}{2}\beta\right)$, $W_0^{(m)} = 1 - \alpha^2$ and $W_i^{(m)} = W_i^{(c)} = \frac{1}{2L\alpha^2}\beta$, β is a tunable parameter used to include prior distribution. The transformed sigma-point vector is then used to predict the measurements using the measurement model:

$$Y_{k|k-1} = h\left[\Psi_{k|k-1}\right] \tag{25}$$

The expected measurement $\hat{y}_k^-$ and square-root covariance of $\tilde{y}_k = y_k - \hat{y}_k^-$ (called the innovation) are given by the unscented transform expressions just as for the process model:

$$\hat{y}_k^- = \sum_{i=0}^{(m)} W_i^{(m)}\, Y_{i,k|k-1} \tag{26}$$

$$S_{\tilde{y}k} = qr\left\{\left[\sqrt{W_1^{(c)}}\left(Y_{1:2L,k|k-1} - \hat{y}_k\right)\sqrt{R_k^v}\right]\right\} \tag{27}$$

$$S_{\tilde{y}k} = cholupdate\left\{S_{\tilde{y}k},\ Y_{0,k} - \hat{y}_k,\ W_0^{(c)}\right\} \tag{28}$$

In an attempt to find out how much to adjust the predicted state mean and covariance based on the actual measurement, the Kalman gain matrix K_k is calculated as follows:

$$P_{Zkyk} = \sum_{i=0}^{2L} W_i^{(c)}\left[\Psi_{i,k|k-1} - \hat{Z}_k^-\right]\left[Y_{i,k|k-1} - \hat{y}_k^-\right] \tag{29}$$

$$K_K = P_{Zkyk}/S_{\tilde{y}k}^T/S_{\tilde{y}k} \tag{30}$$

Finally, the state mean and covariance are updated using the actual measurement and the Kalman gain matrix:

$$\hat{z}_k = \hat{z}_k^- + K_k(y_k - \hat{y}_k^-) \tag{31}$$

$$U = K_k S_{\tilde{y}k} \tag{32}$$

$$S_k = cholupdate\left\{S_k^-,\ U, -1\right\} \tag{33}$$

http://dx.doi.org/10.5772/63728

where R^w is the process noise covariance, R^v is the measurement noise covariance, chol is Cholesky method of matrix factorization, *qr* is QR matrix decomposition and cholupdate is a Cholesky factor updating.

The SRUKF technique is summarized in Algorithm 3.

3.4. SRCDKF method

Like the SRUKF, the matrix square-root S_k will be propagated directly, avoiding the computational complexity to refactorize at each time step in the CDKF. The SRCDKF is initialized with a state mean vector and the square root of a covariance.

$$\hat{z}_0 = E[z_0] And\ S_0 = chol\{E[(z_0 - \hat{z}_0)(z_0 - \hat{z}_0)']\} \tag{34}$$

After the Cholesky factorization we obtain the sigma-point:

$$\Psi_{k-1} = [\hat{z}_{k-1} \quad \hat{z}_{k-1} + hS_{k-1} \quad \hat{z}_{k-1} - hS_{k-1}] \tag{35}$$

The sigma-point vector is then gone through the nonlinear process system, which predicts the current attitude based on each sigma-point.

$$\Psi_{k/k-1} = f[\Psi_{k-1}] \tag{36}$$

The estimated state mean and square-root covariance are calculated from the transformed sigma-point using,

$$\hat{z}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1} \tag{37}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c1)}}\left(\Psi_{1:L,k|k-1} - \Psi_{L+1:2L,k|k-1}\right)\sqrt{W_1^{(c2)}}\left(\Psi_{1:L,k|k-1} + \Psi_{L+1:2L,k|k-1} - 2\Psi_{0,k|k-1}\right)\right]\right\} \tag{38}$$

where $W_i^{(c_1)} = \frac{1}{4h^2}$, $W_i^{(c_2)} = \frac{h^2-1}{4h^4}$, $W_0^{(m)} = \frac{h^2-L}{h^2}$ and $W_i^{(m)} = \frac{1}{2h^2}$. The next step, the sigma-point for measurement update is calculated as,

$$\Psi_{k|k-1} = [\hat{z}_{k|k-1} \quad \hat{z}_{k|k-1} + hS_{k|k-1} \quad \hat{z}_{k|k-1} - hS_{k|k-1}] \tag{39}$$

The transformed sigma-point vector is then used to predict the measurements using the measurement model:

$$Y_{k|k-1} = h\left(\Psi_{k|k-1}\right) \tag{40}$$

The expected measurement $\hat{y}_k^-$ and square-root covariance of $\tilde{y}_k = y_k - \hat{y}_k^-$ (called the innovation) are given by expressions just as for the process model:

$$\hat{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{k|k-1} \tag{41}$$

$$S_{\tilde{y}_k} = qr\left\{\left[\sqrt{W_1^{(c_1)}}\left(Y_{1:L:2L,k|k-1} - Y_{L+1:2L,k|k-1}\right)\sqrt{W_1^{(c_2)}}\left(Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1} - 2Y_{0,k|k-1}\right)\right]\right\} \tag{42}$$

In an attempt to find out how much to adjust the predicted state mean and covariance based on the actual measurement, the Kalman gain matrix K_k is calculated as follows:

$$P_{z_k y_k} = W_1^{(c_1)} S_k^- \left[Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1}\right]^T \tag{43}$$

$$K_k = P_{z_k y_k} / S_{\tilde{y}_k}^T / S_{\tilde{y}_k} \tag{44}$$

Then, the state mean and covariance are updated using the actual measurement and the Kalman gain matrix is:

$$\hat{z}_k = \hat{z}_k^- + K_k(y_k - \hat{y}_k^-) \tag{45}$$

$$U = K_k S_{\tilde{y}_k} \tag{46}$$

$$S_k = cholupdate\{S_k^-, U, -1\} \tag{47}$$

The SRCDKF technique is summarized in Algorithm 4.

3.5. ISPKF

In order to achieve superior performance of the statical linearization methods in terms of efficiency and accuracy, the ISPKFs have been developed. These filters include IUKF, ICDKF, ISRUKF and ISRCDKF. The major difference between the ISPKFs and the noniterated SPKFs is shown in the step where the updated state estimation is calculated using the predicted state and the observation. Instead of relying on the predicted state, the observation equation is relinearized over times by iterating an approximate maximum a posteriori estimate, so the state estimate will be more accurate.

http://dx.doi.org/10.5772/63728

3.5.1. IUKF

The difference between the UKF and the IUKF consists in the iteration strategy.

After generating the prediction and the update steps, and getting both the state estimate $\hat{z}_k$ and the covariance matrix P_k, an iteration loop is set up with the following initializations:

$\hat{z}_{k,0} = \hat{z}_k^-$, $P_{k,0} = P_k^-$, $\hat{z}_{k,1} = \hat{z}_k$, $P_{k,1} = P_k$ and $j = 2$ with j is the jth iterate.

In this loop, for each j, new sigma-points are generated in the same way as the standard UKF

$$\Psi_{k,j} = \left[\hat{z}_{k,j-1} \quad \hat{z}_{k,j-1} + \sqrt{(L+\lambda)P_{k,j-1}} \quad \hat{z}_{k,j-1} - \sqrt{(L+\lambda)P_{k,j-1}}\right] \tag{48}$$

Algorithm 3: SRUKF algorithm

- *Initialization step:* $\hat{z}_0 = E\left[z_0\right]$

$$And\ S_0 = chol\left\{E\left[\left(z_0 - \hat{z}_0\right)\left(z_0 - \hat{z}_0\right)'\right]\right\}$$

- *Prediction step:*

$$\Psi_{k-1} = \left[\hat{z}_{k-1} \qquad \hat{z}_{k-1} + hS_{k-1} \qquad \hat{z}_{k-1} - hS_{k-1}\right]$$

$$\Psi_{k/k-1} = f\left[\Psi_{k-1}\right]$$

$$\hat{Z}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c)}}\left(\Psi_{1:2L,k/k-1} - \hat{Z}_k^-\right)\sqrt{R^w}\right]\right\}$$

$$S_k^- = cholupdate\left\{S_k^-, \Psi_{0,k} - \hat{Z}_k^-, W_0^{(c)}\right\}$$

$$Y_{k|k-1} = h\left[\Psi_{k|k-1}\right]$$

$$\hat{y}_k^- = \sum_{i=0}^{(m)} W_i^{(m)} Y_{i,k|k-1}$$

- *Estimation (update) step:*

$$S_{\tilde{y}_k} = qr\left\{\left[\sqrt{W_1^{(c)}}\left(Y_{1:2L,k|k-1} - \hat{y}_k\right)\sqrt{R_k^v}\right]\right\}$$

$$S_{\tilde{y}_k} = cholupdate\left\{S_{\tilde{y}_k}, Y_{0,k} - \hat{y}_k, W_0^{(c)}\right\}$$

$$P_{Zkyk} = \sum_{i=0}^{2L} W_i^{(c)}[\Psi_{i,k|k-1} - \hat{Z}_k^-][Y_{i,k|k-1} - \hat{y}_k^-]$$

$$K_K = P_{Zkyk} / S_{\tilde{y}k}^T / S_{\tilde{y}k}$$

$$\hat{z}_k = \hat{z}_k^- + K_k\left(y_k - \hat{y}_k^-\right)$$

$$U = K_k S_{\tilde{y}k}$$

$$S_k = cholupdate\left\{S_k^-, U, -1\right\}$$

Return the augmented state estimation $\hat{z}_k$

Algorithm 4: SRCDKF algorithm

- *Initialization step:* $\hat{z}_0 = E[z_0]$

$$And\ S_0 = chol\left\{E\left[\left(z_0 - \hat{z}_0\right)\left(z_0 - \hat{z}_0\right)'\right]\right\}$$

- *Prediction step:*

$$\Psi_{k-1} = \left[\hat{z}_{k-1} \quad \hat{z}_{k-1} + hS_{k-1} \quad \hat{z}_{k-1} - hS_{k-1}\right]$$

$$\Psi_{k/k-1} = f\left[\Psi_{k-1}\right]$$

http://dx.doi.org/10.5772/63728

$$z_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c1)}}\left(\Psi_{1:L,k|k-1} - \Psi_{L+1:2L,k|k-1}\right)\sqrt{W_1^{(c_2)}}\left(\Psi_{1:L,k|k-1} + \Psi_{L+1:2L,k|k-1} - 2\Psi_{0,k|k-1}\right)\right]\right\}$$

$$\Psi_{k|k-1} = \left[\hat{z}_{k|k-1} \qquad \hat{z}_{k|k-1} + hS_{k|k-1} \qquad \hat{z}_{k|k-1} - hS_{k|k-1}\right]$$

$$Y_{k|k-1} = h(\Psi_{k|k-1})$$

$$\hat{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{k|k-1}$$

- *Estimation(update) step:*

$$S_{\bar{y}_k} = qr\left\{\left[\sqrt{W_1^{(c_1)}}\left(Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1}\right)\sqrt{W_1^{(c_2)}}\left(Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1} - 2Y_{0,k|k-1}\right)\right]\right\}$$

$$P_{z_k y_k} = \sqrt{W_1^{(c_1)}} S_k^-\left[Y_{1:L,k|k-1} - Y_{L+1:2L,k|k-1}\right]^T$$

$$K_k = P_{z_k y_k} / S_{\bar{y}_k}^T / S_{\bar{y}_k}$$

$$\hat{z}_k = \hat{z}_k^- + K_k(y_k - \hat{y}_k^-)$$

$$U = K_k S_{\bar{y}_k}$$

$$S_k = cholupdate\left\{S_k^-, U, -1\right\}$$

Return the augmented state estimation $\hat{z}_k$

Then the prediction step and the update step are executed as follows:

$$\hat{z}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,j} \tag{49}$$

$$Y_{k,j} = h\left[\Psi_{i,j}\right] \tag{50}$$

where $Y_{k,j}$ represents the ith component of Y_j

$$\hat{y}_{k,j}^{-} = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,j} \tag{51}$$

$$P_{\tilde{y}_{k,j}} = \sum_{i=0}^{2L} W_i^{(c)} \left[Y_{i,j} - \hat{y}_{k,j}^{-}\right] \left[Y_{i,j} - \hat{y}_{k,j}^{-}\right]^T \tag{52}$$

$$P_{z_{k,j} y_{k,j}} = \sum_{i=0}^{2L} W_i^{(c)} \left[\Psi_{i,,j} - \hat{z}_{k,j}^{-}\right] \left[Y_{i,j} - \hat{y}_{k,j}^{-}\right]^T \tag{53}$$

$$K_{k,j} = P_{z_{k,j} y_{k,j}} P_{\tilde{y}_k,j}^{-1} \tag{54}$$

$$\hat{z}_{k,j} = \hat{z}_{k,j}^{-} + g.K_{k,j}(y_k - \hat{y}_{k,j}^{-}) \tag{55}$$

$$P_{k,j} = P_{k,j}^{-} - K_{k,j} P_{\tilde{y}_{k,j}} K_{k,j}^T \tag{56}$$

Those steps are repeated many times until a following inequality is not satisfied.

$$\hat{z}_{k,j}^T P_{k,j-1}^T + \tilde{y}_{k,j}^T R_k^{-1} \tilde{y}_{k,j} < \tilde{y}_{k,j-1}^T R_k^{-1} \tilde{y}_{k,j-1} \; or \; j < N \tag{57}$$

The IUKF is summarized in Algorithm 5.

3.5.2. ICDKF

The iterated sigma-point methods have the ability to provide accuracy over other estimation methods since it relinearizes the measurement equation by iterating an approximate maximum a posteriori estimate around the updated state, instead of relying on the predicted state.

In the ICDKF, the prediction step is calculated as the standard CDKF and we get $\hat{z}_k^{-}$ and P_k^{-}.

Then the sigma-point in measurement updating is calculated as follows:

$$\Psi_{(k|k-1)} = \left[\hat{z}_{(k|k-1\ )} \quad \hat{z}_{(k|k-1)} + h\sqrt{P_k^{-}} \quad \hat{z}_{(k|k-1\ )} - h\sqrt{P_k^{-}}\ \right] \tag{58}$$

After that, the initialization $z_0 = \hat{z}_k^{-}$ is set up and then the iteration step is executed, so the following equations are repeated m times.

http://dx.doi.org/10.5772/63728

$$\Psi_j = \begin{bmatrix} z_j & z_j + h\sqrt{P_k^-} & z_j - h\sqrt{P_k^-} \end{bmatrix} \tag{59}$$

$$Y_j = h\left[\Psi_j\right] \tag{60}$$

$$\hat{y}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,j} \tag{61}$$

$$P_{\bar{y}_k} = \sum_{i=0}^{2L} W_i^{(c_1)} \left(Y_{i,j} - Y_{L+i,j}\right)^2 + W_i^{(c_2)} \left(Y_{i,j} + Y_{L+i,j} - 2Y_{0,j}\right)^2 \tag{62}$$

$$P_{z_k y_k} = \sqrt{W_1^{(c_1)} P_z} \left[Y_{1:L,j} - Y_{L+1:2L,j}\right]^T \tag{63}$$

$$K_{k,j} = P_{z_k y_k} P_{\bar{y}_k}^{-1} \tag{64}$$

$$\hat{z}_{k,j} = \hat{z}_k^- + K_{k,j}(y_k - \hat{y}_{k,j}^-) \tag{65}$$

$$P_k = P_k^- - K_{k,j} P_{\bar{y}_k} K_{k,j}^T \tag{66}$$

The algorithm of the ICDKF is summarized in Algorithm 6.

3.5.3. ISRUKF

The ISRUKF has the same principle as the IUKF. After executing the standard SRUKF, an iteration loop is started. The predicted estimated state $\left(z_k^-, \hat{z}_k\right)$ and the predicted and estimated covariance matrix (S_k^-, S_k) obtained through the prediction and the update steps will be the initialization inputs for the iteration loop ($\hat{z}_{k,0} = \hat{z}_k^-$, $S_{k,0} = S_k^-$ and $\hat{z}_{k,1} = \hat{z}_k$, $S_{k,1} = S_k$). Also let j=2 where j is the jth iteration.

In the iteration loop, and for each j, the new sigma-point vector is generated as follows:

$$\Psi_{k,j} = [\hat{z}_{k,j-1} \quad \hat{z}_{k,j-1} + hS_{k,j-1} \quad \hat{z}_{k,j-1} - hS_{k,j-1}] \tag{67}$$

Then, the prediction and the update steps are executed as follows:

$$\Psi_j = f(\Psi_{k,j}) \tag{68}$$

$$\hat{z}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,j} \tag{69}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c)}}\left(\Psi_{1:2L,j} - \hat{Z}_{k,j}^-\right)\sqrt{R^w}\right]\right\} \tag{70}$$

$$S_k^- = cholupdate\left\{S_{k,j}^-,\ \Psi_{i,j} - \hat{Z}_{k,j}^-,\ W_1^{(c)}\right\} \tag{71}$$

$$Y_{i,j} = h\left[\Psi_{i,j}\right] \tag{72}$$

$$\hat{y}_{k,j}^- = \sum_{i=0}^{(m)} W_i^{(m)}\, Y_{i,j} \tag{73}$$

$$S_{\tilde{y}k,j} = qr\left\{\left[\sqrt{W_1^{(c)}}\left(Y_{1:2L,j} - \hat{y}_{k,j}\right)\sqrt{R_k^v}\right]\right\} \tag{74}$$

$$S_{\tilde{y}k,j} = cholupdate\left\{S_{\tilde{y}k,j}, Y_{0,j} - \hat{y}_{k,j},\ W_0^{(c)}\right\} \tag{75}$$

$$P_{Zkyk} = \sum_{i=0}^{2L} W_i^{(c)}\left[\Psi_{i,j} - \hat{Z}_{k,j}^-\right]\left[Y_{i,j} - \hat{y}_{k,j}^-\right] \tag{76}$$

$$K_{K,j} = P_{Zkyk}/S_{\tilde{y}k,j}^T/S_{\tilde{y}k,j} \tag{77}$$

$$\hat{z}_{k,j} = \hat{z}_{k-j}^- + K_{k,j}\left(y_k - \hat{y}_{k,j}^-\right) \tag{78}$$

$$U = K_{k,j}S_{\tilde{y}k,j} \tag{79}$$

$$S_{k,j} = cholupdate\left\{S_{k,j-1}^-, U, -1\right\} \tag{80}$$

The equations in the iterative loop are repeated m times.

The ISRUKF algorithm is summarized in Algorithm 7.

3.5.4. ISRCDKF

The ISRCDKF has the ability to provide accuracy over other SRCDKF since it relinearizes the measurement equation by iterating an approximate maximum a posteriori estimate around the updated state, instead of relying on the predicted state.

The algorithm of the ISRCDKF consists of generating the prediction step as the standard SRCDKF, then applying *m* iterations over the update step described as follows:

$$\Psi_j = \left[z_j \quad z_j + hS_k^- \quad z_j - hS_k^-\right] \tag{81}$$

$$Y_j = h\left(\Psi_j\right) \tag{82}$$

$$\hat{y}_{k,j}^{-} = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,j} \tag{83}$$

$$S_{\tilde{y}_k} = qr\left\{\left[\sqrt{W_1^{(c_1)}}\left(Y_{1:L,j} - Y_{L+1:2L,j}\right)\sqrt{W_1^{(c_2)}}\left(Y_{1:L,j} - Y_{L+1:2L,j} - 2Y_{0,j}\right)\right]\right\} \tag{84}$$

$$P_{z_k y_k} = W_1^{(c_1)} S_k^{-}\left[Y_{1:L,j} - Y_{L+1:2L,j}\right]^T \tag{85}$$

$$K_{k,j} = P_{z_k y_k} / S_{\tilde{y}_k}^t / S_{\tilde{y}_k} \tag{86}$$

$$\hat{z}_{k,j} = \hat{z}_k^{-} + K_{k,j}\left(y_k - \hat{y}_{k,j}^{-}\right) \tag{87}$$

$$U = K_{k,j} S_{\tilde{y}_k} \tag{88}$$

$$S_k = cholupdate\{S_k^{-}, U, -1\} \tag{89}$$

The ISRCDKF algorithm is summarized in Algorithm 8.

In the next section, the SPKF method performances will be assessed and compared to ISPKF methods. The performances of UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF methods will be evaluated through a simulation example with two comparative studies in terms of estimation accuracy, convergence and execution times.

4. Simulation results

4.1. State and parameter estimations for a second-order LTI system

Consider a second-order LTI described by the following state variable,

$$\dot{x}_k = Ax_k + BU_1(k) + v_k \tag{90}$$

where v_k is a Gaussian process noise $(v_k; 0, 10^{-1})$, and $A = \begin{pmatrix} 1.9223 & -0.9604 \\ 1 & 0 \end{pmatrix}$ is a matrix with scalar parameter $B = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix}$.

Algorithm 5: IUKF algorithm

- *Initialization step:*

$$Z_0 = E\left[z_0\right]$$

$$P_{z_0} = \left[(z - \hat{z}_0)(z - \hat{z}_0)^T \right]$$

- *Prediction step:*

Generate the UKF prediction step and return $\hat{z}_k^-$ *and* P_k^-

- *Estimation (update) step:*
- *Generate the UKF update step and return* P_k *and* $\hat{z}_k$
- *Iteration: Let* $\hat{z}_{k,0} = \hat{z}_k^-, \quad P_{k,0} = P_k^-$

And $\hat{z}_{k,1} = \hat{z}_k, \quad P_{k,1} = P_k$

Also let $g = 1$ and $j = 2$.

$$\Psi_{k,j} = \left[\hat{z}_{k,j-1} \quad \hat{z}_{k,j-1} + \sqrt{(L+\lambda)P_{k,j-1}} \quad \hat{z}_{k,j-1} - \sqrt{(L+\lambda)P_{k,j-1}} \right]$$

$$\hat{z}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,j}$$

$$Y_{k,j} = h\left[\Psi_{i,j} \right]$$

$$\hat{y}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,j}$$

$$P_{\tilde{y}_{k,j}} = \sum_{i=0}^{2L} W_i^{(c)} \left[Y_{i,j} - \hat{y}_{k,j}^- \right] \left[Y_{i,j} - \hat{y}_{k,j}^- \right]^T$$

$$P_{z_{k,j} y_{k,j}} = \sum_{i=0}^{2L} W_i^{(c)} \left[\Psi_{i,j} - \hat{z}_{k,j}^- \right] \left[Y_{i,j} - \hat{y}_{k,j}^- \right]^T$$

$$K_{k,j} = P_{z_{k,j} y_{k,j}} P_{\tilde{y}_{k,j}}^{-1}$$

$$\hat{z}_{k,j} = \hat{z}_{k,j}^- + g.K_{k,j}(y_k - \hat{y}_{k,j}^-)$$

$$P_{k,j} = P_{k,j}^- - K_{k,j} P_{\tilde{y}_{k,j}} K_{k,j}^T$$

http://dx.doi.org/10.5772/63728

Define these equations:

$$\hat{y}_{k,j} = h(\hat{z}_{k,j})$$

$$\tilde{z}_{k,j} = \hat{z}_{k,j} - \hat{z}_{k,j-1}$$

$$\tilde{y}_{k,j} = y_k - \hat{y}_{k,j}$$

If the inequality is fulfilled

$$\hat{z}_{k,j}^T P_{k,j-1}^T + \tilde{y}_{k,j}^T R_k^{-1} \tilde{y}_{k,j} < \tilde{y}_{k,j-1}^T R_k^{-1} \tilde{y}_{k,j-1}$$

And $j < N$ *then set* $g = \eta \cdot g, \quad j = j + 1$, *and return to the iterated loop.*

Otherwise set

$\hat{z}_k = \hat{z}_{k,j}$ *And* $P_k = P_{k,j}$ *Return the augmented state estimation* $\widehat{z}_k$

Algorithm 6: ICDKF algorithm

- *Initialization step:*

$$Z_0 = E\left[z_0\right]$$

$$P_{z_0} = \left[(z - \hat{z}_0)(z - \hat{z}_0)^T\right]$$

- *Prediction step:*

$$\Psi_{k-1} = \left[\hat{z}_{k-1} \quad \hat{z}_{k-1} + h\sqrt{P_z} \quad \hat{z}_{k-1} - h\sqrt{P_z}\right]$$

$$\Psi_{k|k-1} = f(\Psi_{k-1})$$

$$\hat{z}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1}$$

$$P_k^- = \sum_{i=0}^{2L} W_i^{(c_1)} (\Psi_{i,k|k-1} - \Psi_{L+i,k|k-1})^2 + W_i^{(c_2)} (\Psi_{i,k|k-1} + \Psi_{L+i,k|k-1} - 2\Psi_{0,k|k-1})^2$$

$$\Psi_{(k|k-1)} = \left[\hat{z}_{(k|k-1)} \; \hat{z}_{(k|k-1)} + h\sqrt{P_k^-} \; \hat{z}_{(k|k-1)} - h\sqrt{P_k^-} \right]$$

- *Estimation(update) step:*

$$z_0 = \hat{z}_k^-$$

- *Iteration: for j=0,.., m*

$$\Psi_j = \left[z_j \quad z_j + h\sqrt{P_k^-} \quad z_j - h\sqrt{P_k^-} \right]$$

$$Y_j = h\left[\Psi_j \right]$$

$$\hat{y}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,j}$$

$$P_{\bar{y}_k} = \sum_{i=0}^{2L} W_i^{(c_1)} \left(Y_{i,j} - Y_{L+i,j} \right)^2 + W_i^{(c_2)} \left(Y_{i,j} + Y_{L+i,j} - 2Y_{0,j} \right)^2$$

$$P_{z_k y_k} = \sqrt{W_1^{(c_1)} P_z} \left[Y_{1:L,j} - Y_{L+1:2L,j} \right]^T$$

$$K_{k,j} = P_{z_k y_k} P_{\bar{y}_k}^{-1}$$

End

$$\hat{z}_{k,j} = \hat{z}_k^- + K_{k,j} (y_k - \hat{y}_{k,j}^-)$$

$$P_k = P_k^- - K_{k,j} P_{\bar{y}_k} K_{k,j}^T$$

http://dx.doi.org/10.5772/63728

Return the augmented state estimation $\widehat{z}_k$

Algorithm 7: ISRUKF algorithm

- *Initialization step:* $z_0 = E[z_0]$

$$S_0 = chol\left\{E\left[(z_0 - \hat{z}_0)(z_0 - \hat{z}_0)'\right]\right\}$$

- *Prediction step:*
- *Generate the SRUKF prediction step and return* $\hat{z}_k^-$ *and* S_k^-
- *Estimation (update) step:*

Generate the SRUKF update step and return S_k *and* $\hat{z}_k$

- *Iteration: Let* $\hat{z}_{k,0} = \hat{z}_k^-$, $S_{k,0} = S_k^-$ and $\hat{z}_{k,1} = \hat{z}_k$, $S_{k,1} = S_k$. Also let *j*=2

$$\Psi_{k,j} = \left[\hat{z}_{k,j-1} \quad \hat{z}_{k,j-1} + hS_{k,j-1} \quad \hat{z}_{k,j-1} - hS_{k,j-1}\right]$$

$$\Psi_j = f(\Psi_{k,j})$$

$$\hat{z}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,j}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c)}}\left(\Psi_{1:2L,j} - \hat{Z}_{k,j}^-\right)\sqrt{R^w}\right]\right\}$$

$$S_k^- = cholupdate\left\{S_{k,j}^-, \Psi_{i,j} - \hat{Z}_{k,j}^-, W_1^{(c)}\right\}$$

$$Y_{i,j} = h\left[\Psi_{i,j}\right]$$

$$\hat{y}_{k,j}^- = \sum_{i=0}^{(m)} W_i^{(m)} Y_{i,j}$$

$$S_{\tilde{y}k,j} = qr\left\{\left[\sqrt{W_1^{(c)}}\left(Y_{1:2L,j} - \hat{y}_{k,j}\right)\sqrt{R_k^v}\right]\right\}$$

$$S_{\tilde{y}k,j} = cholupdate\left\{S_{\tilde{y}k,j}, Y_{0,j} - \hat{y}_{k,j}, W_0^{(c)}\right\}$$

$$P_{Zkyk} = \sum_{i=0}^{2L} W_i^{(c)} \left[\Psi_{i,j} - \hat{Z}_{k,j}^{-}\right]\left[Y_{i,j} - \hat{y}_{k,j}^{-}\right]$$

$$K_{K,j} = P_{Zkyk} / S_{\tilde{y}k,j}^{T} / S_{\tilde{y}k,j}$$

$$\hat{z}_{k,j} = \hat{z}_{k-j}^{-} + K_{k,j}\left(y_k - \hat{y}_{k,j}^{-}\right)$$

$$U = K_{k,j} S_{\tilde{y}k,j}$$

$$S_{k,j} = cholupdate\left\{S_{k,j-1}^{-}, U, -1\right\}$$

End

$$U = K_{k,m} S_{\tilde{y}_k}$$

$$S_k = cholupdate\left\{S_{k,m}, U, -1\right\}$$

Return the augmented state estimation $\hat{z}_k$

Algorithm 8: ISRCDKF algorithm

- *Initialization step:* $\hat{z}_0 = E\left[z_0\right]$

$$S_0 = chol\left\{E\left[\left(z_0 - \hat{z}_0\right)\left(z_0 - \hat{z}_0\right)'\right]\right\}$$

- *Prediction step:*

$$\Psi_{k-1} = \left[\hat{z}_{k-1} \quad \hat{z}_{k-1} + hS_{k-1} \quad \hat{z}_{k-1} - hS_{k-1}\right]$$

$$\Psi_{k/k-1} = f\left[\Psi_{k-1}\right]$$

http://dx.doi.org/10.5772/63728

$$z_k^- = \sum_{i=0}^{2L} W_i^{(m)} \Psi_{i,k|k-1}$$

$$S_k^- = qr\left\{\left[\sqrt{W_1^{(c_1)}}\left(\Psi_{1:L,k|k-1} - \Psi_{L+1:2L,k|k-1}\right)\sqrt{W_1^{(c_2)}}\left(\Psi_{1:L,k|k-1} + \Psi_{L+1:2L,k|k-1} - 2\Psi_{0,k|k-1}\right)\right]\right\}$$

$$\Psi_{(k|k-1)} = \left[\hat{z}_{(k|k-1)} \quad \hat{z}_{(k|k-1)} + hS_{(k|k-1)} \quad \hat{z}_{(k|k-1)} - hS_{(k|k-1)}\right]$$

- *Estimation(update) step:*

$$z_0 = \hat{z}_k^-$$

- *Iteration: for j=0: m*

$$\Psi_j = \left[z_j \quad z_j + hS_k^- \quad z_j - hS_k^-\right]$$

$$Y_j = h\left(\Psi_j\right)$$

$$\hat{y}_{k,j}^- = \sum_{i=0}^{2L} W_i^{(m)} Y_{i,j}$$

$$S_{\tilde{y}_k} = qr\left\{\left[\sqrt{W_1^{(c_1)}}\left(Y_{1:L,j} - Y_{L+1:2L,j}\right)\sqrt{W_1^{(c_2)}}\left(Y_{1:L,j} - Y_{L+1:2L,j} - 2Y_{0,j}\right)\right]\right\}$$

$$P_{z_k y_k} = W_1^{(c_1)} S_k^- \left[Y_{1:L,j} - Y_{L+1:2L,j}\right]^T$$

$$K_{k,j} = P_{z_k y_k} / S_{\tilde{y}_k}^t / S_{\tilde{y}_k}$$

$$\hat{z}_{k,j} = \hat{z}_k^- + K_{k,j}\left(y_k - \hat{y}_{k,j}^-\right)$$

$$U = K_{k,j} S_{\tilde{y}_k}$$

$$S_k = cholupdate\left\{S_k^-, U, -1\right\}$$

End $U = K_{k,m} S_{\tilde{y}_k}$

$$S_k = cholupdate\left\{S_{k,m}, U, -1\right\}$$

Return the augmented state estimation $\hat{z}_k$

The nonstationary observation model is given by,

$$y_k = Cx_k + DU_2(k) + n_k \tag{91}$$

where $C = [1\ \ 0]$ and $D = \begin{bmatrix} 0 \\ 0.2 \end{bmatrix}$. The observation noise n_k is a Gaussian noise $\mathcal{N}(n_k; 0, 3.10^{-1})$. Given only the noisy observations $y_{k'}$ the different filters were used to estimate the underlying clean state sequence x_k for $k = 1...100$.

4.1.1. Generation of dynamic data

It must be noted that this simulated state is assumed to be noise-free. They are contaminated with Gaussian noise. Given noisy observations $y_{k'}$ the various KFs were used to estimate the clean state sequence $x_k = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ for k = 1...100. **Figure 1** shows the changes in the state variable x_1.

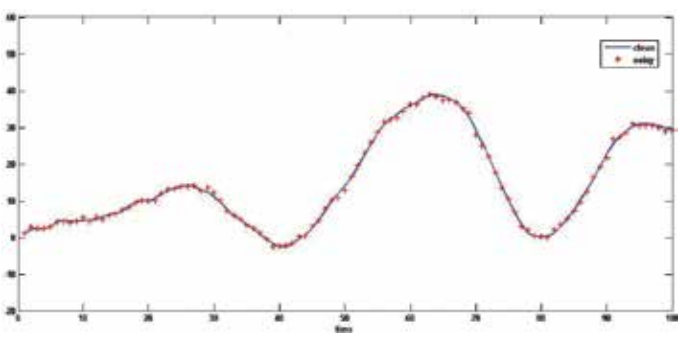

Figure 1. Simulated data used in estimation: state variable (x_1).

Here, the number of sigma-points is fixed to 9 for all the techniques ($L = 4$). The process noise $(v_k; 0, 10^{-1})$ was added. The observation noise is $\mathcal{N}(n_k; 0, 3.10^{-1})$. The initial value of the state vector is $x_0 = [1\ \ 0]'$.

4.1.2. Comparative study: estimation of state variables from noisy measurements

The purpose of this study is to compare the estimation accuracy of UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF methods when they are utilized to estimate the state variable of the system. Hence, it is considered that the state vector to be estimated $z_k = x_k$ and the model parameters P_1, P_2 are assumed to be known. The simulation results for state estimations of state variable x_k using UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF methods are shown in **Figures 2** and **3**, respectively. Also, the performance comparison of the state estimation techniques in terms of RMSE and execution times is presented in **Table 1**.

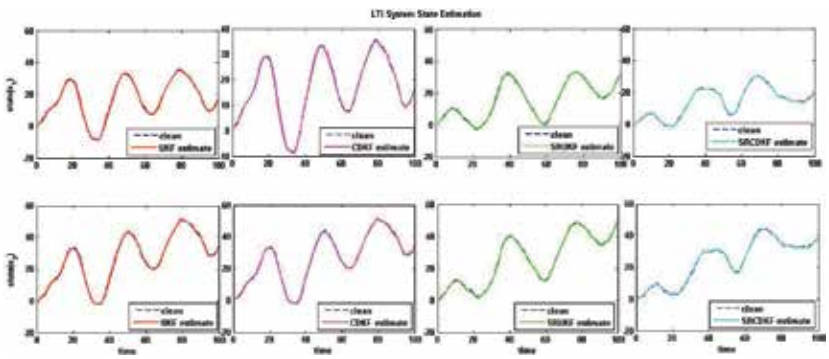

Figure 2. Estimation of state variables using various state estimation techniques (UKF, CDKF, SRUKF and SRCDKF).

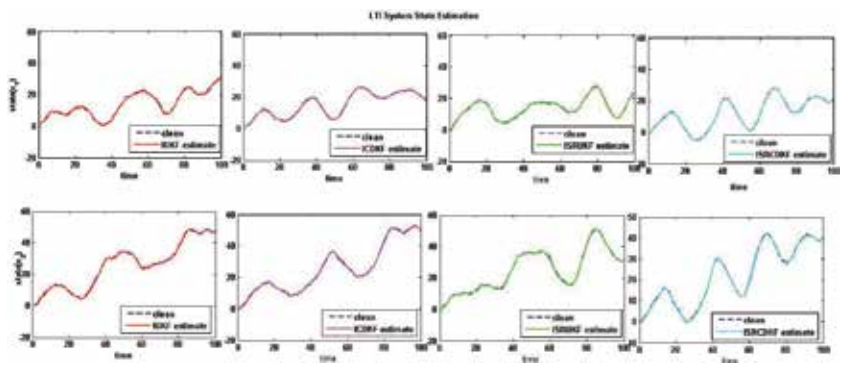

Figure 3. Estimation of state variables using various state estimation techniques (IUKF, ICDKF, ISRUKF and ISRCDKF).

Technique	x_1 (RMSE)	x_2 (RMSE)	Time execution(s)	Technique	x_1 (RMSE)	x_2 (RMSE)	Time execution(s)
UKF	0.3539	0.4658	0.3577	IUKF	0.3342	0.4341	0.5952
CDKF	0.3512	0.4583	0.3367	ICDKF	0.3265	0.4315	0.4351
SRUKF	0.3495	0.4590	0.3354	ISRUKF	0.3254	0.4256	0.5803
SRCDKF	0.3324	0.4593	0.2586	ISRCDKF	0.3121	0.4213	0.4229

Table 1. Comparison of state estimation techniques.

It is easily observed from **Figures 2** and **3** as well as **Table 1** that the ISRCDKF method achieves a better accuracy than the other methods.

4.1.3. Comparative study: simultaneous estimation of state variables and model parameters

The state variables and parameters are estimated and performed using the state estimation techniques UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF. The results of estimation for the model parameters using the estimation techniques (UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF) are shown in **Figures 4** and **5**, respectively. It can be seen from the results presented in **Figures 4** and **5** that the IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF methods outperform the UKF method, and that the ISRCDKF shows relative improvement over all other techniques. These results confirm the results obtained in the first comparative study, where only the state variable is estimated. The advantages of the ISRCDKF over the other techniques can also be seen through their abilities to estimate the model parameters.

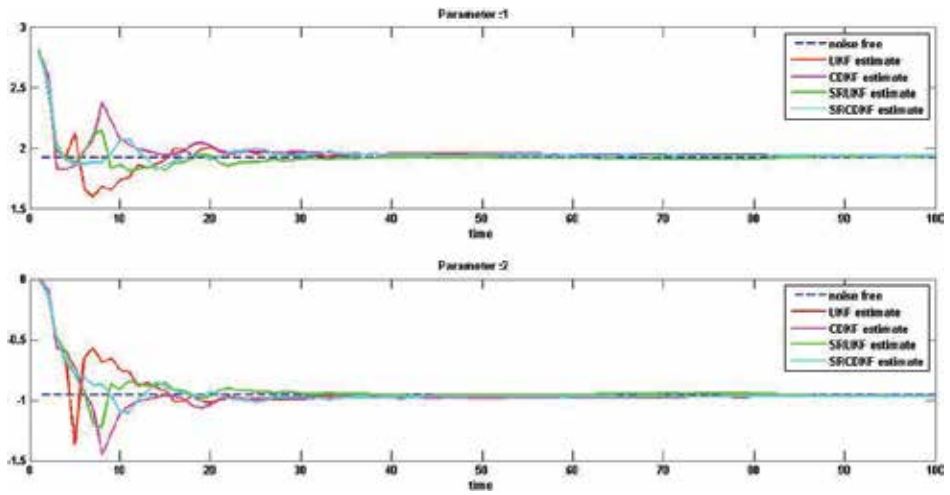

Figure 4. Estimation of the model parameters (P_1, P_2) using UKF, CDKF, SRUKF and SRCDKF.

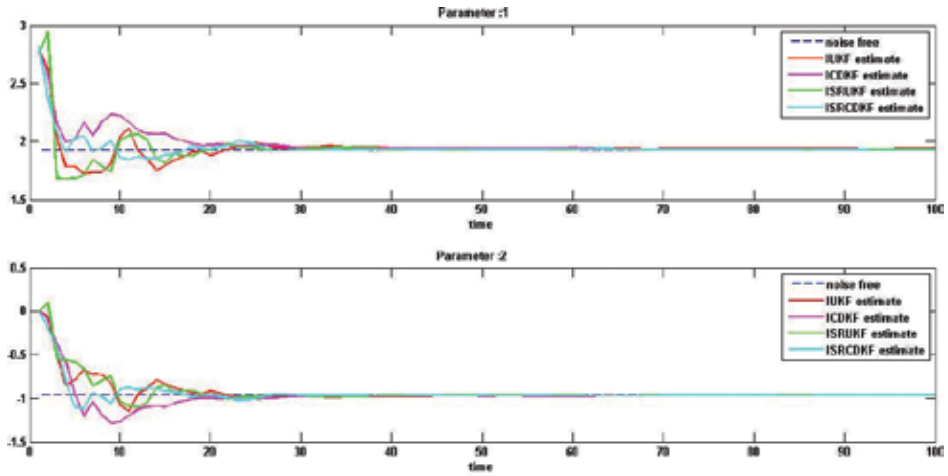

Figure 5. Estimation of the model parameters (P_1, P_2) using IUKF, ICDKF, ISRUKF and ISRCDKF.

4.1.3.1. Root Mean Square Error analysis

The effects of the practical challenges on the performances of the UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF for state and parameter estimation are investigated in the next section.

4.1.3.1.1. Effect of number of state and parameter to estimate on the estimation RMSE

To study the effect of the number of states and parameters to be estimated on the estimation performances of UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF, the estimation performance is analyzed for different numbers of estimated states and parameters. Here, we will consider two cases, which are summarized below. In all cases, it is assumed that the state x_k is measured.

Case 1: the state x_k along with the first parameter P_1 will be estimated.

Case 2: the state x_k along with the two parameters P_1 and P_2 will be estimated.

The estimation of the state variables and parameter(s) for these two cases is performed using UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF, and the simulation results for the state variables and the model parameters for the two cases are shown in **Tables 2** and **3**. For example, for case 1, **Table 2** compares the estimation RMSEs for the two state variables x_k (with respect to the noise-free data) and the mean of the estimated parameter P_1 at steady state (i.e., after convergence of parameter(s)) using the estimation methods. **Tables 2** and **3** also present similar comparisons for cases 1 and 2, respectively.

Technique	x_1 (RMSE)	x_2 (RMSE)	P_1 (mean)	Time execution (s)	Technique	x_1 (RMSE)	x_2 (RMSE)	P_1 (mean)	Time execution (s)
UKF	0.4221	0.5418	2.2453	0.3906	IUKF	0.3854	0.5093	1.9826	0.6963
CDKF	0.4192	0.5205	2.2232	0.3696	ICDKF	0.3827	0.4920	1.9786	0.5160
SRUKF	0.4063	0.4978	2.2228	0.3835	ISRUKF	0.3757	0.4748	1.9661	0.6798
SRCDKF	0.3970	0.4943	2.1858	0.3420	ISRCDKF	0.3737	0.4720	1.9297	0.5154

Table 2. Root mean square errors of estimated state variables and mean of estimated parameter: case 1.

The results also show that the number of parameters to estimate affects the estimation accuracy of the state variables. In other words, for all the techniques the estimation RMSE of x_k increases from the first comparative study (where only the state variables are estimated) to case 1 (where the states and one parameter P_1 is estimated) to case 2 (where the states and two parameters, P_1 and P_2, are estimated). For example, the RMSEs obtained using ISRCDKF for x_1 in the first comparative study and cases 1–2 of the second comparative study are 0.3121, 0.3737 and 0.3846, respectively, which increase as the number of estimated parameters increases (see **Tables 2** and **3**). This observation is valid for the other state estimation techniques.

It can also be shown from **Tables 2** and **3** that, for all the techniques, estimating more model parameters affects the estimation accuracy. The ISRCDKF method, however, still provides advantages over other methods in terms of the estimation accuracy.

Technique	x_1 (RMSE)	x_2 (RMSE)	P_1 (mean)	P_2 (mean)	Technique	x_1 (RMSE)	x_2 RMSE)	P_1 (mean)	P_2 (mean)
UKF	0.1962	0.6590	1.9484	-0.9798	IUKF	0.4056	0.4927	19408	-0.9721
CDKF	0.4170	0.4932	1.9482	-0.9786	ICDKF	0.4012	0.4908	1.9389	-0.9720
SRUKF	0.4133	0.4977	1.9481	-0.9776	ISRUKF	0.3989	0.4843	1.9342	-0.9677
SRCDKF	0.4090	0.4956	1.9436	-0.9741	ISRCDKF	0.3846	0.4875	1.9305	-0.9486

Table 3. Root mean square errors of estimated state variables and mean of estimated parameter: case 2.

4.1.3.1.2. Effect of noise content on the estimation RMSE

It is assumed that a noise is added to the state variable. In order to show the performance of the estimation algorithms in the presence of noise, three different measurement noise values, 10^{-1}, 10^{-2} and 10^{-3}, are considered. The simulation results of estimating the state x_k using the UKF, IUKF, CDKF, ICDKF, SRUKF, ISRUKF, SRCDKF and ISRCDKF methods when the noise levels vary in $\{10^{-1}, 10^{-2} \text{ and } 10^{-3}\}$ are shown in **Table 4**.

Noise levels		UKF	CDKF	SRUKF	SRCDKF	IUKF	ICDKF	ISRUKF	ISRCDKF
10^{-1}	x_1	0.3539	0.3512	0.3495	0.3324	0.3342	0.3265	0.3254	0.3121
	x_2	0.4658	0.4593	0.4590	0.4593	0.4341	0.4315	0.4256	0.4213
10^{-2}	x_1	0.1293	0.1264	0.1208	0.1174	0.1134	0.1095	0.1075	0.1066
	x_2	0.3564	0.3493	0.3474	0.3457	0.3440	0.3371	0.3355	0.3314
10^{-3}	x_1	0.0460	0.0454	0.0448	0.0446	0.0436	0.0415	0.0394	0.0376
	x_2	0.3426	0.3360	0.3188	0.3062	0.2989	0.2918	0.2875	0.2830

Table 4. Root mean square errors (RMSEs) of the estimated states for different noise levels.

In other words, for the estimation techniques, the estimation RMSEs of x_k increase from the first comparative study (noise value = 10^{-1}) to case (where the noise value = 10^{-3}). For example, the RMSEs obtained using ISRCDKF for x_1 where the noise level in $\{10^{-1}, 10^{-2}$ and $10^{-3}\}$ are 0.3121, 0.1066 and 0.0376, which increase as the noise variance increases (refer to **Table 4**).

5. Conclusions

In this chapter, various SPKF-based methods are used to estimate nonlinear state variables and model parameters. They are compared for the estimation performance in two comparative studies. In the first comparative study, the state variables are estimated from noisy measurements of these variables, and the several estimation methods are compared by estimating the RMSE with respect to the noise-free data. In the second comparative study, of the state variables as well as that the model parameters are estimated. Comparing the performances of the several state estimation extensions, the impact of the number of estimated model parameters on the convergence and accuracy of these methods is also evaluated. The results of the second comparative study show that, for all the techniques, estimating more model parameters affects the estimation accuracy as well as the convergence of the estimated states and parameters. The iterated square-root central difference Kalman method, however, still provides advantages over other methods in terms of the estimation accuracy, convergence and execution times.

Acknowledgements

This work was made possible by NPRP grant NPRP7-1172-2-439 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

Author details

Marwa Chaabane[1,2], Imen Baklouti[2,4], Majdi Mansouri[1*], Nouha Jaoua[2], Hazem Nounou[1], Mohamed Nounou[3], Ahmed Ben Hamida[2] and Marie-France Destain[4]

*Address all correspondence to: majdi.mansouri@qatar.tamu.edu

1 Electrical and Computer Engineering Program, Texas A&M University at Qatar, Doha, Qatar

2 Advanced Technologies for Medicine and Signals, National Engineering School of Sfax, Tunisia

3 Chemical Engineering Program, Texas A&M University at Qatar, Doha, Qatar

4 Biosystems Engineering Department, GxABT, University of Liege, Gembloux, Belgium

References

[1] W. Guo. Dynamic state-space models. Journal of Time Series Analysis. 2003;24(2):149–158.

[2] G. Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. Journal of Computational. 1996;5(1):1–25.

[3] J. H. Kotecha and P. M. Djurić. Gaussian sum particle filtering for dynamic state space models. Acoustics, Speech and Signal Processing. 2001;6(IEEE, 2001): 3465–3468.

[4] S.-G. Cao, N. W. Rees, and G. Feng. Analysis and design of fuzzy control systems using dynamic fuzzy-state space. Fuzzy Systems, IEEE Transactions on. 1999;7(2):192–200.

[5] C.-J. Kim and C. R. Nelson. State-space models with regime switching: classical and Gibbs-sampling approaches with. MIT Press, Cambridge. 1999;2.

[6] A. Nabavi-Niaki and M. R. Iravani. Steady-state and dynamic models of unified power flow controller (UPFC) for power. Power Systems, IEEE Transactions. 1996;11(4):1937–1943.

[7] M. Mansouri, O. Ilham, H. Snoussi, and C. Richard. Adaptive quantized target tracking in wireless sensor networks. Wireless Networks. 2011;17(7):1625–1639.

[8] M. Mansouri, M. Mohamed-Seghir, H. N. Nounou, M. N. Nounou, and H. Abu-Rub. Bayesian methods for time-varying state and parameter estimation in induction machines. International Journal of Adaptive Control and Signal Processing, 2015, 29(7), 905-924.

http://dx.doi.org/10.5772/63728

[9] M. Mansouri, H. N. Nounou, M. N. Nounou, and A. A. Datta. State and parameter estimation for nonlinear biological phenomena modeled by S-systems. Digital Signal Processing. 2014;28:1–17.

[10] M. Mansouri, H. Nounou, and M. Nounou. Modeling of nonlinear biological phenomena modeled by S-systems. Mathematical Biosciences. 2014;249:75–91.

[11] A. Simoglou, E. Martin, and A. Morris. Statistical performance monitoring of dynamic multivariate processes using state space modelling. Computers and Chemical Engineering. 2002;26(6):909–920.

[12] M. Mansouri, H. Nounou, and M. Nounou. State estimation of a chemical reactor process model-a comparative study. IEEE 10th International Multi-Conference, in Systems, Signals & Devices (SSD), 2013. 2013:1–6.

[13] M. Mansouri, B. Dumont, and M.-F. Destain. Modeling and prediction of time-varying environmental data using advanced Bayesian methods. Exploring Innovative and Successful Applications of Soft Computing. 2013:112.

[14] H. van der Kooij, R. Jacobs, B. Koopman, and F. van der Helm. An adaptive model of sensory integration in a dynamic. Biological Cybernetics. 2001;84(2):2103–2115.

[15] M. Mansouri and M.-F. Destain. Predicting biomass and grain protein content using Bayesian methods. Stochastic Environmental Research and Risk Assessment. 2015;29(4):1167–1177.

[16] G. L. Plett. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation. Journal of Power Sources. 2004;134(2):277–292.

[17] D. Dochain. State and parameter estimation in chemical and biochemical processes: a tutorial. Journal of Process. 2003;13(8):801–818.

[18] T. A. Wenzel, K. Burnham, M. Blundell, and R. Williams. Dual extended Kalman filter for vehicle state and parameter. Vehicle System Dynamics. 2006;44(2):153–171.

[19] G. Evensen. The ensemble Kalman filter for combined state and parameter estimation. Control Systems, IEEE. 2009;29(3):83–104.

[20] J. Ching, J. L. Beck, and K. A. Porter. Bayesian state and parameter estimation of uncertain dynamical systems. Probabilistic Engineering Mechanics. 2006;21(1):81–96.

[21] H. Moradkhani, S. Sorooshian, H. V. Gupta, and P. R. Houser. Dual state–parameter estimation of hydrological models using ensemble Kalman filter. Advances in Water Resources. 2005;28(2):135–147.

[22] P. Moireau, D. Chapelle, and P. Le Tallec. Joint state and parameter estimation for distributed mechanical systems. Computer Methods in Applied Mechanics and Engineering. 2008;197(6):659–677.

[23] S. Aborhey and D. Williamson. State and parameter estimation of microbial growth processes. Automatica. 1978;14(5):493–498.

[24] S. Rao, M. Buss, and V. Utkin. Simultaneous state and parameter estimation in induction motors using first-and second order sliding modes. Industrial Electronics, IEEE Transactions on. 2009;56(9):3369–3376.

[25] A. V. Savkin and I. R. Petersen. Recursive state estimation for uncertain systems with an integral quadratic constraint. IEEE Transactions on Automatic Control. 1995;40(6): 1080–1083.

[26] D. Simon. Optimal state estimation: Kalman, H8, and nonlinear approaches. John Wiley and Sons, 2006, pages 552, 978-0-471-70858-2.

[27] M.S. Grewal, and A.P. Andrews, Kalman filtering: theory and practice using MATLAB. Theory and Practice, 2001, 5(5), pp.634-6.

[28] V. Aidala. Parameter estimation via the Kalman filter. IEEE Trans. on Automatic Control. 1977;22(3):471–472.

[29] L. Matthies, T. Kanade, and R. Szeliski. Kalman filter-based algorithms for estimating depth from image sequences. International Journal of Computer Vision. 1989;3(3):209–238.

[30] S. Julier and J. Uhlmann. New extension of the Kalman filter to nonlinear systems. Proceedings of SPIE. 1997;3(1):182–193.

[31] L. Ljung. Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems. IEEE Trans. 1979;24(1):36–50.

[32] Y. Kim, S. Sul, and M. Park. Speed sensorless vector control of induction motor using extended Kalman filter. IEEE Trans. on Industrial Applications. 1994;3(5):1225–1233.

[33] E. Wan and R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000. 2000;153–158.

[34] R. Van Der Merwe and E. Wan. The square-root unscented Kalman filter for state and parameter-estimation. In Acoustics, Speech, and Signal Processing, 2001. Proceedings (ICASSP'01) 2001 IEEE International Conference. 2001;6:3461–3464.

[35] S. Sarkka. On unscented Kalman filtering for state estimation of continuous-time nonlinear systems. IEEE Trans Automatic Control. 2007;52(9):1631–1641.

[36] M. Mansouri, H. Nounou, M. Nounou, and A. A. Datta. Modeling of nonlinear biological phenomena modeled by S-systems. In: Biomedical Engineering and Sciences (IECBES), 2012 IEEE EMBS Conference on IEEE. 2012;305–310.

http://dx.doi.org/10.5772/63728

[37] J. Zhu, N. Zheng, Z. Yuan, Q. Zhang, X. Zhang, and Y. He. A slam algorithm based on the central difference Kalman filter. In: Intelligent Vehicles Symposium, 2009 IEEE. 2009;123–128.

[38] M. M. Andreasen. Non-linear DSGE models and the central difference Kalman filter. Journal of Applied Econometrics. 2013;28(6):929–955.

[39] S. Holmes, G. Klein, and D. W. Murray. A square root unscented Kalman filter for visual monoslam. In: Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on. IEEE. 2008;3710–3716.

[40] M. Huang, W. Li, and W. Yan. Estimating parameters of synchronous generators using square-root unscented Kalman filter. Electric Power Systems Research. 2010;80(9):1137–1144.

[41] M. NøRgaard, N. K. Poulsen, and O. Ravn. New developments in state estimation for nonlinear systems. Automatica. 2000;36(11):1627–1638.

[42] R. Zhan and J. Wan. Iterated unscented Kalman filter for passive target tracking. Aerospace and Electronic Systems, IEEE Transactions on. 2007;43(3):1155–1163.

[43] XK. Yiyu. Iterated unscented kalman filter. Journal of Huazhong University of Science and Technology (Nature Science Edition) 2007, 11, 006.

[44] L.M. Sibley, Gabe, Gaurav S. Sukhatme, and Larry Matthies. "The Iterated Sigma Point Kalman Filter with Applications to Long Range Stereo." In Robotics: Science and Systems, vol. 8, no. 1, pp. 235-244. 2006.

[45] G. Sibley, G. Sukhatme, and L. Matthies. The iterated sigma point Kalman filter with applications to long range stereo. In Robotics: Science and Systems. 2006;8(1):235–244.

[46] M. Mansouri, O. Avci, H. Nounou, and M. Nounou. A comparative assessment of nonlinear state estimation methods for structural health monitoring. Model Validation and Uncertainty Quantification. 2015;3:45–54.

[47] J. Heming, M. Hongwei, Gao Wei, Che Yanting. Application of ISRCDKF in SINS initial alignment with large azimuth misalignment angle. AISS (Advances in Information Sciences and Service Sciences). 2013;5:746–755.

[48] R. Van Der Merwe, E. A. Wan, S. Julier et al. Sigma-point Kalman filters for nonlinear estimation and sensor-fusion: applications to integrated navigation. In: Proceedings of the AIAA Guidance, Navigation & Control Conference. 2004;16–19.

Chapter 7

An Introduction to Ensemble-Based Data Assimilation Method in the Earth Sciences

Youmin Tang, Zheqi Shen and Yanqiu Gao

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64718

Abstract

In this chapter, the ensemble-based data assimilation methods are introduced, including their developments, applications and existing concerns. These methods include both traditional methods such as Kalman filter and its derivatives and some advanced algorithms such as sigma-point Kalman filters and particle filters. Emphasis is placed on the challenges of applying these methods onto high-dimensional systems in the earth sciences.

Keywords: data assimilation, Kalman filter, EnOI, EnKF, particle filter

1. Introduction

In this chapter, we will talk about the modelling and simulation using both observed data and numerical models, that is, the observations will be incorporated into numerical models for optimal modelling and simulation. In statistics, this is called state-space estimation. In the earth science, it is called data assimilation. For example, a strict definition of data assimilation in atmospheric and oceanic sciences is the process to estimate the state of a dynamic system such as atmospheric and oceanic flow by combining the observational and model forecast data [1].

In general, assimilation methods can be classified into two categories: variational and sequential. This chapter is a tutorial on the sequential data assimilation methods such as ensemble Kalman filter (EnKF) and its variants. A brief introduction of the particle filter (PF) is also provided in this chapter.

This tutorial places emphasis on the rationale behind each method, including: (i) the principle for deriving the algorithm; (ii) the basic assumptions of each method; (iii) the connection and relation between different methods (e.g. extended Kalman filter (EKF) and EnKF, EnKF and sigma-point Kalman filters (SPKF), etc.); and (iv)the advantage and deficiency of each method.

This chapter has been written and organized through teaching for under-/graduatestudents in earth science courses. It can also be a good reference to researchers in the field of modelling and data assimilation.

2. The general framework of several assimilation approaches

Intuitionally, one might think that an optimal simulation scheme is to directly replace model variables by observations during numerical integrations. Such a direct replacement is usually not correct since observations are not perfect and contain errors. A simple replacement will introduce observation errors into models, and ignore possible impact of observation errors on model behaviours, easily resulting in imbalance of model dynamics and physics. Thus, the application of observations into numerical models must consider both model and observation errors that play a critical role in the assimilation process.

We will start to display the assimilation concept by a simple example. A detail introduction can be found in [2].

For an unknown true state value, denoted by T_t, there are two samples, denoted by T_1(e.g. model simulation) and T_2(observation), which have the errors ϵ_1 and ϵ_2, respectively. Thus, we have

$$T_1 = T_t + \epsilon_1, \tag{1}$$

$$T_2 = T_t + \epsilon_2. \tag{2}$$

We assume the measurement or observation is unbiased, and the variances of errors are known, i.e. $E(\epsilon_1) = E(\epsilon_2) = 0$, $Var(\epsilon_1) = \sigma_1^2$, $Var(\epsilon_2) = \sigma_2^2$. The question here is to seek an optimal estimate, denoted by T_a (called analysis in the assimilation field), for T_t using T_1 and T_2. This optimal estimate is the central issue of data assimilation.

There are several methods for this solution, as demonstrated below.

2.1. Least-squares method

Assume the analysis is a linear combination of both T_1 and T_2, that is, $T_a = a_1T_1 + a_2T_2$. Due to the assumption that both T_1 and T_2 are unbiased, T_a should be unbiased, i.e. $E(T_a) = E(T_t)$, so $a_1E(T_1) + a_2E(T_2) = E(T_t)$, and then $a_1 + a_2 = 1$. The best (optimal) estimate should minimize the variance of T_a as below:

$$\begin{aligned}\sigma_a^2 &= E\left[T_a - T_t\right]^2 = E\left[a_1 T_1 + a_2 T_2 - T_t\right]^2 = E\left[a_1\left(T_1 - T_t\right) + a_2\left(T_2 - T_t\right)\right]^2 \\ &= E\left(a_1^2\epsilon_1^2 + a_2^2\epsilon_2^2 + 2a_1a_2\epsilon_1\epsilon_2\right) = a_1^2\sigma_1^2 + \left(1 - a_1\right)^2\sigma_2^2,\end{aligned} \tag{3}$$

here, we assumed that the errors of T_1 and T_2 are uncorrelated, i.e. $E(\epsilon_1\epsilon_2)=0$. To minimize σ_a^2, let $\partial\sigma_a^2/\partial a_1 = 0$, thus

$$a_1 = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \tag{4}$$

Namely,

$$T_a = a_1T_1 + (1 - a_1)T_2 = T_1 + \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}(T_2 - T_1). \tag{5}$$

Using Eq. (5), the variance of T_a could be minimized.

2.2. Variational approach

In general, assimilation methods can be classified into two categories: variational and sequential. Variational methods such as three-dimensional variational (3D-Var) method and four-dimensional variational (4D-Var) method [3, 4] are batch methods, whereas sequential methods such as Kalman filter (KF) [5] belong to the estimation theory.

They both have had great success. The European Centre for Medium-Range Weather Forecasts (ECMWF) introduced the first 4D-Var method into the operational global analysis system in November 1997 [6–8]. The ensemble Kalman filter (EnKF) was first introduced into the operational ensemble prediction system by Canadian Meteorological Centre (CMC) in January 2005 [9].

This chapter is a tutorial of the ensemble-based sequential data assimilation methods, such as EnKF and its variants. However, we will briefly demonstrate the idea of variational assimilation by the above example.

First, a cost function should be defined for variational assimilation approach. For this simple example, we define the cost function as below:

$$J(T) = \frac{1}{2}\left[\frac{(T - T_1)^2}{\sigma_1^2} + \frac{(T - T_2)^2}{\sigma_2^2}\right] \tag{6}$$

$$T = a_1T_1 + a_2T_2. \tag{7}$$

The solution is to seek an analysis T_a, determined by a_1 and a_2, leading to the cost function minimum, i.e. $J(T_a) = \min\{J(T)\}$. Obviously, we have $\partial J(T)/\partial a_1 = 0$ and $\partial J(T)/\partial a_2 = 0$. Substitute with (6), it is

$$\frac{\partial J(T)}{\partial a_1} = \frac{T - T_1}{\sigma_1^2}\frac{\partial T}{\partial a_1} + \frac{T - T_2}{\sigma_2^2}\frac{\partial T}{\partial a_1} = 0 \tag{8}$$

Eq. (7) leads to $\frac{\partial T}{\partial a_1} = T_1$. Thus, the solution of (8), denoted by T_a, satisfies

$$T_a = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}T_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}T_2. \tag{9}$$

The above is a simple example of the 3D variational assimilation approach, where we only consider the analysis error (cost function) for a single time. However, in many cases, we need to consider the error growth during a period, i.e. the sum of errors during the period, in the cost function Eq. (6). For example, the cost function of 4D-Var is defined as below:

$$J(T) = \frac{1}{2}\sum_{n=1}^{N}[\frac{(T(t_n) - T_1(t_n))^2}{\sigma_1^2} + \frac{(T(t_n) - T_2(t_n))^2}{\sigma_2^2}]. \tag{10}$$

Meanwhile $T(t_n)$ follows a dynamical model, saying $T(t_n) = \int_{t_0}^{t_n} F(T(t))dt = M_n(T(t_0))$, where F is a nonlinear dynamical model, M_n is the integral operator and t_0 is the initial time. Thus, the cost function value of (10) is only determined by the initial condition. Namely, the objective here is to seek optimal initial condition $T(t_0)$ that enables (10) minimum, i.e. minimizing (10) subject to dynamical model F. This is a standard conditional extreme problem that can be solved by Lagrange multiplier approach. However, the complexity of dynamical model excludes the possibility to get the analytical solution. We have to solve the minimum problem with aid of numerical methods, e.g. Newton conjugate gradient method. All of numerical methods require the gradient value $\frac{\partial J}{\partial T_0}$ for solution.

Again, it is almost impossible for obtaining analytical solution of $\frac{\partial J}{\partial T_0}$ due to the complexity of F. Usually researchers get the gradient value numerically using an approach of tangent linear and adjoint models. The details on tangent linear and adjoint models can be found in relevant references as cited above. It should be noticed that it is very difficult, even intractable sometimes, to construct tangent linear and adjoint models in some cases. Thus, more and more researchers have started to apply sequential assimilation methods instead of 4D-Var in recent years. Next, we will introduce the concept of the sequential assimilation method using the above example.

http://dx.doi.org/10.5772/64718

2.3. Bayesian approach

Assume T_1 and σ_1 are the mean value and standard deviation of the model prediction that implies a prior probability distribution of truth T,

$$p(T) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(T-T_1)^2}{2\sigma_1^2}} \tag{11}$$

Obviously, this is a Gaussian distribution function, which can be denoted by N(T_1, σ_1) Given the observation T_2 and its standard deviation σ_2, the posterior distribution of the truth can be expressed by Bayes' theorem:

$$p(T|T_2) = \frac{p(T_2|T)p(T)}{p(T_2)} \propto \frac{1}{\sqrt{2\pi}\sigma_2} e^{\frac{(T-T_2)^2}{2\sigma_2^2}} \frac{1}{\sqrt{2\pi}\sigma_1} e^{\frac{(T-T_1)^2}{2\sigma_1^2}}. \tag{12}$$

$p(T_2)$ was ignored in (12) since it is independent of T, and usually plays as a normalization factor. The likelihood function $p(T_2|T)$ describes the probability that the observation becomes T_2 given an estimation of T. It is commonly assumed to be Gaussian $N(T, \sigma_2)$. The object here is to estimate the truth by maximizing the posterior probability $p(T|T_2)$(namely, we ask the truth to occur as much as possible—maximum probability). Maximizing $p(T|T_2)$ is equivalent to maximizing the logarithm of the right item of (12), i.e.

$$\begin{aligned} \log(p(T|T_2)) &= \log\left(\frac{1}{\sqrt{2\pi}\sigma_2}\right) - \frac{(T-T_2)^2}{2\sigma_2^2} + \log\left(\frac{1}{\sqrt{2\pi}\sigma_1}\right) - \frac{(T-T_1)^2}{2\sigma_1^2} \\ &= \text{const} - \frac{1}{2}\left[\frac{(T-T_2)^2}{\sigma_2^2} + \frac{(T-T_1)^2}{\sigma_1^2}\right]. \end{aligned} \tag{13}$$

Obviously, the maximum of $p(T|T_2)$ occurs at the minimum of the second item on the right-hand side of (13), i.e. the minimum of the cost function J of (6). Thus, under the assumption of Gaussian distribution, maximizing a posterior probability (Bayesian approach) is equivalent to minimizing cost function (variational assimilation approach). Further, if the model F is linear and the probability distribution is Gaussian, it can be further proved that the Kalman filter is equivalent to 4D-Var adjoint assimilation method.

3. Optimal interpolation (OI) and Kalman filter (KF)

3.1. Optimal interpolation

The most special case in data assimilation is that the forecast model is linear and the errors are Gaussian. The solution among sequential methods to this case is provided by Kalman filter. Typically, the Kalman filter applies to the below state-space model:

$$x_{t+1} = Mx_t + \eta_t, \tag{14}$$

$$y_t = Hx_t + \zeta_t, \tag{15}$$

where M and H are linear operators of model and measurement, respectively. x is model state and y is the observation, and the subscript implies the time step. η_t and ζ_t are the model errors and observational errors, respectively, which have variance: $var(\eta_t) =< \eta_t, \eta_t^T >= Q, var(\eta_t) =< \zeta_t, \zeta_t^T >= R$. The objective here is to estimate model state x using y, making it close to true state (unknown) as much as possible.

Assuming the estimate of model state x^a at a time step is a linear combination of model forecast x^b and observation y^o, i.e. the filter itself is linear, so

$$x^a = x^b + K\left[y^o - Hx^b\right]. \tag{16}$$

Eq. (16) is the standard expression of Kalman filter. K is called Kalman gain that determines the optimal estimate and $y^o - Hx^b$ is called the innovation. An analysis step is essentially to determine the increment to the forecast by combining the Kalman gain and the innovation. Before deriving K, we denote the covariance matrix of the analysis error ϵ^a by P^a, i.e. $P^a = < \epsilon^a , (\epsilon^a)^T >$, where $\epsilon^a = x^a - x^{tr}$ and x^{tr} is the true value of model state. Similarly, observed errors and forecast errors are defined by $\epsilon^o = y^o - Hx^{tr}$ and $\epsilon^b = x^b - x^{tr}$, respectively. It should be noticed that the forecast error ϵ^b is different from the model error ζ_t that is a systematic bias. Also, we denote $B = < \epsilon^b, (\epsilon^b)^T >$ as the background (forecast) error covariance and $R = < \epsilon^o, (\epsilon^o)^T >$ as the observational error covariance. It is also assumed that the observation error is not related to forecast error, so $< \epsilon^b, (\epsilon^o)^T > = < \epsilon^o, (\epsilon^b)^T > = 0$.

Clearly, we are seeking for K that can lead to P^a minimum. Subtracting x^{tr} on both sides of Eq. (16) leads to the below equation:

$$x^a - x^{tr} = x^b - x^{tr} + K\left[y^o - Hx^b + Hx^{tr} - Hx^{tr}\right]. \tag{17}$$

http://dx.doi.org/10.5772/64718

Namely,

$$\epsilon^a = \epsilon^b + K\left(\epsilon^o - H\epsilon^b\right), \tag{18}$$

And

$$\begin{aligned} P^a &= E\left[\epsilon^b + K\left(\epsilon^o - H\epsilon^b\right)\right]\left[\epsilon^b + K\left(\epsilon^o - H\epsilon^b\right)\right]^T \\ &= E\left[\epsilon^b\left(\epsilon^b\right)^T + \epsilon^b\left(\epsilon^o - H\epsilon^b\right)^T K^T + K\left(\epsilon^o - H\epsilon^b\right)\left(\epsilon^b\right)^T + K\left(\epsilon^o - H\epsilon^b\right)\left(\epsilon^o - H\epsilon^b\right)^T K^T\right] \\ &= B - BH^T K^T - KHB + K\left(R + HBH^T\right)K^T. \end{aligned} \tag{19}$$

Here, we used $B = B^T$. The optimal estimate asks the trace of P^a minimum, namely, $\partial\left[trace\left(P^a\right)\right]/\partial K = 0$. It can be computed that

$$K = BH^T(HBH^T + R)^{-1}. \tag{20}$$

Substitute into (13)

$$P^a = B - BH^T K^T - KHB + BH^T(HBH^T + R)^{(-1)}(R + HBH^T)K^T = (I - KH)B. \tag{21}$$

Here, we invoked the below properties:

$$\frac{\partial Ax}{\partial x^T} = \frac{\partial x^T A}{\partial x} = A \tag{22}$$

$$\frac{\partial x^T Ax}{\partial x} = \left(A + A^T\right)x \tag{23}$$

$$\frac{\partial A^T x}{\partial x^T} = \frac{\partial x^T A^T}{\partial x} = A^T \tag{24}$$

$$\frac{\partial(\text{trace}[XAX^T])}{\partial x} = X(A + A^T) \tag{25}$$

$$\frac{\partial(\text{trace}[AX^T])}{\partial x} = A^T \tag{26}$$

Thus, we have the optimal estimate filter:

$$x^a = x^b + K\left[y^o - Hx^b\right], \tag{27}$$

$$K = BH^T(HBH^T + R)^{-1}, \tag{28}$$

$$P^a = (I - KH)B. \tag{29}$$

In the estimate (27)–(29), if the background error covariance B is prescribed, the estimate is called optimal interpolation. The OI does not involve state equation (14) and B is unchanged during the entire assimilation process.

3.2. Kalman filter

Now, we consider that B in (28) changes with the assimilation cycle. This is more realistic since the model prediction errors should be expected to decrease with the assimilation.

From Eq. (14), we have

$$x_{t+1}^{tr} = Mx_t^{tr} + \eta_t, \tag{30}$$

$$x_{t+1}^b = E(Mx_t^a + \eta_t) = Mx_t^a \tag{31}$$

Eq. (30) indicates that even the true value is input at a time step, model cannot get a true value for next step due to model bias η_t. Eq. (31) shows a standard procedure for the model prediction of next step starting from the analysis of previous step.

Subtracting (30) from (31) produces

$$\epsilon_{t+1}^b = M\epsilon_t^a - \eta_t, \tag{32}$$

$$B_{t+1} = E\left(\epsilon_{t+1}^b\left(\epsilon_{t+1}^b\right)^T\right) = E[(M\epsilon_t^a + \eta_t)(M\epsilon_t^a + \eta_t)^T] = MP_t^aM^T + Q \tag{33}$$

where $P_t^a = < \epsilon_t^a , (\epsilon_t^a)^T >$ represents the analysis error covariance for time step t. The above equation considers the evolution of the background (prediction) error covariance with the time controlled by the dynamical model operator M. The above equations constitute the framework of Kalman filter (**Table 1**), namely

Analysis step	$x_t^a = x_t^b + K[y^o - Hx_t^b],$
	$K = B_t H^T (HB_t \ H^T + R)^{-1},$
	$P_t^a = (I - KH)B_t,$
Prediction step	$x_{t+1}^b = Mx_t^a,$
	$B_{t+1} = MP_t^a M^T + Q$

Table 1. The Kalman filter.

One Kalman filter cycle consists of two parts, namely, one analysis step (Eqs. (27)–(29)) and one prediction step (Eqs. (31) and (33)). The analysis state x_t^a and covariance P_t^a are treated as initial conditions for the prediction step, until the next observation is available. Sometimes, B_t is denoted by P_t^f in Kalman filter literatures.

3.3. Extended Kalman filter (EKF)

In deriving the Kalman filter, we assume the state model M and measurement model H are both linear. Further, we also assume the error has Gaussian distribution. Therefore, classic KF only works for linear models and Gaussian distribution. If the dynamical model and measurement model are not linear, we cannot directly apply KF. Instead, linearization must be performed prior to apply KF. The linearized version of KF is called extended KF (EKF), which solves the below state-space estimate problem:

$$x_{t+1} = f(x_t) + \eta_t, \tag{34}$$

$$y_t = h(x_t) + \zeta_t, \tag{35}$$

where f and h are nonlinear models, and η_t and ζ_t are additive noises.

The filter is still assumed to be 'linear', i.e.

$$x^a = x^b + K[y^o - h(x^b)] \tag{36}$$

Actually, it is not a linear combination of the forecast x^b and observation y^o if his not linear. However, we just extend the formulation of Eq. (16), and apply it intuitively in nonlinear cases. Ignoring high-order terms, the following holds approximately

$$h(x+\delta x) = h(x) + \frac{\partial h}{\partial x}\delta x = h(x) + H\delta x \tag{37}$$

where H is the linearization of h and $H_{i,j} = \frac{\partial h_i}{\partial x_j}$. So,

$$y^o - h(x^b) = y^o - h(x^{tr} + x^b - x^{tr}) = y^o - h(x^{tr}) - H(x^b - x^{tr}) = \epsilon^o - H\epsilon^b \tag{38}$$

$$x^a = x^b + K(\epsilon^o - H\epsilon^b) \tag{39}$$

Eq. (39) is identical to Eq. (16). Similarly, subtracting x^{tr} on both sides of Eq. (47) leads to the below equation:

$$\epsilon^a = \epsilon^b + K(\epsilon^o - H\epsilon^b) \tag{40}$$

which is the same as Eq. (18). Following the same derivation as that for Eq. (18), we can obtain the equations similar to (27)–(29). Therefore, if the measurement model h is nonlinear, the KF can be still applied with a linearization of h.

Similar to Eqs. (30) and (31), the state model is as below:

$$x_{t+1}^{tr} = f(x_t^{tr}) + \eta_t \tag{41}$$

$$x_{t+1}^{f} = E(f(x_t^a) + \eta_t) = f(x_t^a). \tag{42}$$

Subtracting Eq. (41) from Eq. (42) produces

$$\begin{aligned} \epsilon_{t+1}^f &= f(x_t^a) - f(x_t^{tr}) - \eta_t = f(x_t^a) - f(x_t^{tr} - x_t^a + x_t^a) - \eta_t \\ &= f(x_t^a) - f(x_t^a - \epsilon_t^a) - \eta_t = M\epsilon_t^a - \eta_t \end{aligned} \tag{43}$$

where $M_{i,j} = \frac{\partial f_i}{\partial x_j}$.

Comparing Eq. (31) with Eq. (33), it reveals that Eq. (33) still works. Thus, the EKF can be summarized as below (**Table 2**).

The procedure to perform EKF is similar to that for KF, as listed above. The disparities and similarities between EKF and KF include

i. Kalman gain K has the same form for both, especially the linear or linearized measurement model should be used;

ii. the update equation of model error covariance has the same form, with linear and linearized state model used;

iii. forecast model is different, with KF using linear Eq. (14) and EKF using nonlinear model Eq. (34); and

iv. the filtering algorithm is different, linear measurement model *H* used in KF and nonlinear model *h* in EKF.

It should be noticed that EKF is only an approximate of KF for nonlinear state model.

Analysis step	$x_t^a = x_t^b + K[y^o - h(x_t^b)]$,
	$K = B_t H^T (HB_t H^T + R)^{-1}$,
	$P_t^a = (I - KH)B_t$,
	$H_{i,j} = \frac{\partial h_i}{\partial x_j}$.
Prediction step	$x_{t+1}^f = f(x_t^a)$,
	$B_{t+1} = MP_t^a M^T + Q$
	$M_{i,j} = \frac{\partial f_i}{\partial x_j}$,

Table 2. The extended Kalman filter.

4. Ensemble Kalman filter (EnKF)

4.1. Basics of EnKF

A challenge in EKF is to update background (prediction) error covariance, which requires the linearization of nonlinear model. The linearization of nonlinear model is often difficult technically, and even intractable in some cases, e.g. non-continuous functions existing in

models. Another drawback of EKF is to neglect the contributions from higher-order statistical moments in calculating the error covariance.

To avoid the linearization of nonlinear model, the ensemble Kalman filter (EnKF) was introduced by Evensen [10, 11], in which the prediction error covariance B of Eq. (33) are estimated approximately using an ensemble of model forecasts. The main concept behind the formulation of the EnKF is that if the dynamical model is expressed as a stochastic differential equation, the prediction error statistics, which are described by the Fokker-Flank equation, can be estimated using ensemble integrations ([10, 12]; thus, the error covariance matrix B can be calculated by integrating the ensemble of model states. The EnKF can overcome the EKF drawback that neglects the contributions from higher-order statistical moments in calculating the error covariance. The major strengths of the EnKF include the following:

i. there is no need to calculate the tangent linear model or Jacobian of nonlinear models, which is extremely difficult for ocean (or atmosphere) general circulation models (GCMs);

ii. the covariance matrix is propagated in time via fully nonlinear model equations (no linear approximation as in the EKF); and

iii. it is well suited to modern parallel computers (cluster computing) [13].

EnKF has attracted a broad attention and been widely used in atmospheric and oceanic data assimilation.

Simply saying, EnKF avoids the computation and evolution of the error covariance B as in Eq. (33), and computes *B* using below formula as soon as it is required.

$$B = \frac{1}{N-1}\sum_{i=1}^{N}(x_i^b - \overline{x^b})(x_i^b - \overline{x^b})^T \tag{44}$$

where x_i^b represents the *i*-th member of the forecast ensemble of system state vector at step *t*, and *N* is the ensemble size. The use of Eq. (44) avoids processing *M*, the linearized operator of nonlinear model. However, the measurement function *H* is still linear or linearized while computing the Kalman gain *K*, which causes concern. To avoid the linearization of nonlinear measurement function, Houtekamer and Mitchell [14] wrote Kalman gain by

$$K = BH^T(HBH^T + R)^{-1}, \tag{45}$$

$$BH^T \equiv \frac{1}{N-1}\sum_{i=1}^{N}[x_i^b - \overline{x^b}][h(x_i^b) - \overline{h(x^b)}]^T, \tag{46}$$

$$HBH^T \equiv \frac{1}{N-1}\sum_{i=1}^{N}[h(x_i^b) - \overline{h(x^b)}][h(x_i^b) - \overline{h(x^b)}]^T, \tag{47}$$

where $\overline{h(x^b)} = \frac{1}{N}\sum_{i=1}^{N} h(x_i^b)$. Eqs. (46) and (47) allow direct evaluation of the nonlinear measurement function h in calculating Kalman gain. However, Eqs. (46) and (47) have not been proven mathematically, and only were given intuitionally. Tang and Ambadan argued that Eqs. (46) and (47) approximately hold if and only if $\overline{h(x^b)} = h(\overline{x^b})$ and $x_i^b - \overline{x^b}$ is small for $i = 1, 2, ..., N$ [15]. Under these conditions, Tang et al. argued Eqs. (46) and (47) actually linearize the nonlinear measurement functions h to H [16]. Therefore, direct application of the nonlinear measurement function in Eqs. (46) and (47), in fact, imposes an implicit linearization process using ensemble members. In next section, we will see that Eqs. (46) and (47) can be modified under a rigorous framework.

Thus, the procedures of EnKF are summarized as below (**Table 3**):

1. Imposing perturbations on initial conditions and integrate the model, i.e. $x_{i,1} = f\left(x_0 + \gamma_i\right)$, where $i = 1, 2..., N$(ensemble size) and x_0 is the initial condition.

2. Using $K = BH^T(HBH^T + R)^{-1}$ and Eqs. (46) and (47) to calculate Kalman gain K.

3. Calculating analysis using

$$x_i^a = x_i^b + K\left[y^o + \varepsilon^i - h\left(x_i^b\right)\right], \tag{48}$$

   after K is obtained. It should be noted that each ensemble member produces an analysis; the average of all (N) analyses can be obtained.

4. Using $x_{i,t+1}^b = f(x_i^a)$ to obtain new ensemble members for next round analysis.

5. Repeating Steps 2–4 until the end of assimilation period.

Analysis step	$x_i^a = x_i^b + K[y^o + \varepsilon^i - h(x_i^b)], i = 1, ..., N$
	$K = BH^T(HBH^T + R)^{-1},$
	$BH^T = \frac{1}{N-1}\sum_{i=1}^{N}[x_i^b - \overline{x^b}][h(x_i^b) - \overline{h(x^b)}]^T,$
	$HBH^T = \frac{1}{N-1}\sum_{i=1}^{N}[h(x_i^b) - \overline{h(x^b)}][h(x_i^b) - \overline{h(x^b)}]^T$
Prediction step	$x_{i,t+1}^b = f\left(x_i^a + \gamma_i\right), \quad i = 1, ..., N$

Table 3. The ensemble Kalman filter.

It should be noted that the observation should be treated as a random variable with the mean equal to y^o and covariance equal to R. This is why there is ε_i in Eq. (48). Simply, ε_i is often drawn from a normal distribution $\varepsilon_i \sim N(0, R)$.

From the above procedure, we find that Eq. (44) is not directly applied here. Instead, we use Eqs. (46) and (47) to calculate K. This is because Eqs. (46) and (47) avoid the linearization of nonlinear model and also avoid the explicit expression of matrix B, which is often very large and cannot be written in current computer sources in many realistic problems. The measurement function, h, projecting model space (dimension) to observation space (dimension), greatly reduces the number of dimension.

4.2. Some remarks on EnKF with large dimensional problems

4.2.1. Initial perturbation

The success of EnKF highly depends on the quality of ensemble members produced by initial perturbations. It is impractical to represent all possible types of errors within the ensemble because of the computational cost, the method of generating initial perturbations must be chosen judiciously.

The first issue is the amplitude of initial perturbations. Usually, the following two factors are considered when selecting the amplitude of initial perturbations: the amplitude of observation error and the amplitude of model errors induced by model parameters and uncertainty in model physics. If a model is perfect, the amplitude of the perturbations should be the same as the amplitude of observation errors. This combined error is used to determine the amplitude of perturbations.

When the perturbation amplitude is determined, the practical initial perturbation field generating each ensemble member could be constructed by a normalized pseudorandom field multiplied by the prescribed amplitude. Considering the spatial coherence, the pseudorandom field is red noise as proposed by Evensen [17], summarized as below:

1. Calculate the statistical characteristics for the pseudorandom field to meet its variance of 1 and mean of 0 by solving the following nonlinear equation:

$$e^{-1} = \frac{\sum_{l,p} e^{-2(k_l^2 + r_p^2)/\sigma^2} \cos(k_l r_h)}{\sum_{l,p} e^{-2(k_l^2 + r_p^2)/\sigma^2}}, \tag{49}$$

where $k_l = \frac{2\pi l}{x_n} = \frac{2\pi l}{N_x \Delta x}$, $r_p = \frac{2\pi p}{y_m} = \frac{2\pi p}{N_y \Delta y}$, and N_x and N_y are the number of grid points in the x-axis (lon.) and the y-axis (lat.). The l and p are wavenumbers, changing from 1 to the maximum value of $N/2$ and $M/2$. Δx and Δy are the intervals of two adjacent points, often set to 1, and r_h is the decorrelation length. The purpose of Eq. (49) is to derive σ^2 for the other feature:

http://dx.doi.org/10.5772/64718

$$c^2 = \frac{1}{\Delta k \sum_{l,p} e^{-2(k_l^2 + r_p^2)/\sigma^2}} \tag{50}$$

2. After c and σ^2 are obtained, we can construct a two-dimensional pseudorandom field:

$$W(x_n, y_m) = \sum_{l,p} \frac{c}{\sqrt{\Delta k}} e^{-\frac{\left(k_l^2 + r_p^2\right)}{\sigma^2}} e^{2\pi i \varphi(l,p)} e^{i\left(k_l x_n + r_p y_m\right)} \Delta k. \tag{51}$$

3. While x_n, y_m cover the whole domain, Eq. (51) produces a $N_x * N_y$ two-dimensional random field with spatial coherence structure and the variance of 1 and mean of 0. If the realistic uncertainty (error) has an amplitude β, the perturbation should be βW. Similarly, Eq. (51) is often used for the error perturbation γ_i used in the fourth step of the EnKF procedure.

Sometimes, we need to consider the vertical coherence of pseudorandom fields between adjacent levels in oceanic models. A simple method for this purpose is to construct the pseudorandom field at the kth level ε_k by following equation:

$$\varepsilon_k = \alpha \varepsilon_{k-1} + \sqrt{1-\alpha^2} W_k, \tag{52}$$

where $W_k\left(k = 1, ..., N_z\right)$ is the pseudorandom field at the kth level without considering vertical coherence, constructed using the above method. Initially, for the surface perturbation ($k = 1$), the vertical coherence is not considered, i.e. equals to zero since ε_{k-1} does not exist. Eq. (52) indicates that a practical pseudorandom at the kth level (ε_k) is composed of W_k and ε_{k-1}. As such the ε_k is correlated with ε_{k-1}, i.e. the practical pseudorandom fields of two adjacent levels (ε_{k-1} and ε_k) are coherent with each other. Their correlation or coherent structure is determined by the coefficient $\alpha \in [0, 1]$. Eq. (52) generates a sequence that is white in the vertical direction if $\alpha = 0$(i.e. $\varepsilon_k = W_k$), but a sequence that is perfect correlated in vertical if $\alpha = 1$(i.e. $\varepsilon_k = \varepsilon_{k-1}$). Eq. (52) is also often used to construct random field that is temporally coherent, for example, a continuous random noise that has coherence in time, as used for γ_i in the forecast model [17]. The random noise γ_i in the EnKF procedure can also be replaced by the random noise imposed in model forcing. For example, the random noise is continuously added to wind forcing for oceanic models. Even for some atmospheric models with transition processes, there are inherent random noises making γ_i not necessary. One important criteria for γ_i and the amplitude β is to examine ensemble spread by some sensitivity experiments.

4.2.2. The computational cost of Kalman gain

The Kalman gain K has dimension of $L*m$, where L is the number of model variables and m is the number of observational variables. In many realistic problems, L and m are very large numbers ($m \gg N$, the ensemble size), making the inversion very expensive.

A simple procedure is to rewrite the Kalman gain K, as below:

$$K = \tilde{x}\tilde{x}^T H^T (H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T)^{-1}, \tag{53}$$

where $\tilde{x}$ indicates that the model ensemble predictions removed the ensemble mean ($\tilde{x}_i = [x_i^b - \overline{x^b}]$, for $i = 1, 2, ..., N$). $R = \frac{1}{N}\varepsilon\varepsilon^T$ was invoked here. If we assume the ensemble prediction error ($x^b - \overline{x^{tr}} \approx x^b - \overline{x^b} = \tilde{x}$) is not correlated to observation error, i.e. $\tilde{x}\varepsilon^T = 0$, the following is valid [17]:

$$(H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T) = (H\tilde{x} + \varepsilon)(H\tilde{x} + \varepsilon)^T, \tag{54}$$

where $(H\tilde{x} + \varepsilon)$ has dimension $m*N$. Usually, ensemble size N is much less than m. Using the singular-value decomposition (SVD) technique, we have

$$\left(H\tilde{x} + \varepsilon\right) = U\Sigma V^T \tag{55}$$

Eq. (54) then becomes

$$(H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T) = U\Sigma V^T V\Sigma^T U = U\Sigma\Sigma^T U^T = U\Lambda U^T \tag{56}$$

So,

$$(H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T)^{-1} = U\Lambda^{-1}U^T \tag{57}$$

where U and Λ are the eigenvector and the square of eigenvalues of $(H\tilde{x} + \varepsilon)$. There are N non-zero eigenvalues for $(H\tilde{x} + \varepsilon)$, therefore the dimension is not large, allowing us to efficiently compute the inversion for a global analysis in most practical situations.

4.2.3. Stochastic EnKF and deterministic EnKF

In EnKF introduced in the previous section, the observation assimilated into dynamical model should be treated to be stochastic variable, as expressed by $y^o + \varepsilon^i$ in Eq. (48). It is a must if the classic EnKF algorithm is used. It has been proven that if the EnKF assimilates deterministic

observations (i.e., observation y^o not changed at each ensemble member), the analysis error covariance will be systematically underestimated, typically leading to filter divergence, as indicated by below [11, 18]:

$$P^{a*} = (I - KH)B(I - KH)^T \tag{58}$$

Eq. (58) gives the analysis error covariance if the observed is not perturbed. Comparing Eq. (58) with Eq. (29), a theoretically unbiased estimate, P^{a*} is always less than P^a.

However, the perturbed observation approach (i.e. $y^o + \varepsilon^i$) introduces an additional source of sampling error that reduces analysis error covariance accuracy and increases the probability of understanding analysis error covariance [19, 20]. Thus, an approach that only uses a single observation realization but avoids systematical underestimation of analysis error covariance was pursued. There are several approaches to implement this goal, as summarized by Tippettet al. [20]. Below, we will introduce an approach developed by Whitaker and Hamill [19], called Ensemble squareroot filter (EnSRF).

Denote the deviation of analysis from the analysis mean by $\tilde{x}^a = x^a - \bar{x}^a$, it is easy to write

$$\tilde{x}^a = \tilde{x}^b + \tilde{K}\left[\tilde{y}^o - H\tilde{x}^b\right] \tag{59}$$

where $\tilde{y}^o = y^o - \bar{y}^o$. If a single observation realization is assimilated in all ensemble members, $\tilde{y}^o = 0$ and

$$\tilde{x}^a = \tilde{x}^b - \tilde{K}H\tilde{x}^b = \left(I - \tilde{K}H\right)\tilde{x}^b, \tag{60}$$

$$P^{a*} = \left(I - \tilde{K}H\right)B\left(I - \tilde{K}H\right)^T. \tag{61}$$

We seek a definition for $\tilde{K}$ that will result in an ensemble whose analysis error covariance equals to $(I - KH)B$, i.e.

$$(I - \tilde{K}H)B(I - \tilde{K}H)^T = \left(I - KH\right)B. \tag{62}$$

The solution of Eq. (62) is

$$\tilde{K} = (1 + \sqrt{\frac{R}{HBH^T + R}})^{-1} K. \tag{63}$$

Therefore, EnSRF is summarized as below (**Table 4**):

$\bar{x}^a = \overline{x^b} + K\left[y^o - H\overline{x^b}\right]$
$\tilde{x}^a = \tilde{x}^b - \tilde{K}H\tilde{x}^b$
$x^a = \bar{x}^a + \tilde{x}^a$
$K = BH^T(HBH^T + R)^{-1},,$
$\left[BH^T\right] = \frac{1}{N-1}\sum_{i=1}^{N}[x_i^b - \overline{x^b}][h(x_i^b) - \overline{h(x^b)}]^T$
$HBH^T = \frac{1}{N-1}\sum_{i=1}^{N}[h(x_i^b) - \overline{h(x^b)}][h(x_i^b) - \overline{h(x^b)}]^T$
$\tilde{K} = (1 + \sqrt{\frac{R}{HBH^T + R}})^{-1} K$

Table 4. The analysis scheme of EnSRF.

It should be noted that there are two Kalman gains used in EnSRF, the original K for updating ensemble mean and a new $\tilde{K}$ for updating the anomalies. It indicates that one single observation realization of classic EnKF has the same ensemble analysis mean as stochastic observations.

Initially, the term EnKF refers, in particular, to the stochastic ensemble Kalman filter that requires perturbing the observations. Subsequently, several deterministic EnKFs that avoid the use of perturbed observations were developed, e.g. the ETKF [21], the EAKF [22] and the EnSRF. These filter designs are labelled as variants of the EnKF because they are also based on the Kalman filtering formula and ensemble representations.

http://dx.doi.org/10.5772/64718

4.2.4. Inflation approach

The forecast error covariance is defined by (44)

$$B = \frac{1}{N-1}\sum_{i=1}^{N}(x_i^b - \overline{x^b})(x_i^b - \overline{x^b})^T = \frac{1}{N-1}\tilde{X} * \tilde{X}^T. \tag{64}$$

Eq. (64) is an approximation to B using forecast ensemble. Due to limited computational source, the ensemble size N is often restricted to a small value for many realistic issues. A small ensemble size may cause a very small ensemble spread, causing the approximation of B by Eq. (64), which is seriously underestimated. To solve this problem, B is multiplied by an inflator factor λ (slightly greater than 1). λ is empirically determined, such as some sensitivity experiments, with the typical value of 1.01. λB is used to replace B in EnKF formula. This approach is equivalent to the below approach:

$$x_i^b = \lambda\left(x_i^b - \overline{x^b}\right) + \overline{x^b} \tag{65}$$

4.2.5. Localization of EnKF

When EnKF is applied to high-dimensional atmospheric and oceanic models, the limited ensemble size will cause the estimated correlations to be noisy [11]. When the ensemble size is insufficient, it will produce spurious correlations between distant locations in the background covariance matrix B. Unless they are suppressed, these spurious correlations will cause observations from one location to affect the analysis in locations an arbitrarily large distance away, in an essentially random manner [23]. This needs to be remedied by the localization method.

Another reason for using localization is that the treatment of localization artificially reduces the spatial domain of influence of observations during the update. The localization dramatically reduces the necessary ensemble size, which is very important for operational systems. Two most common distance-based localization methods used in practice are local analysis and covariance localization.

Using local analysis, only measurements located within a certain distance from a grid point will impact the analysis in this grid point. This allows for an algorithm where the analysis is computed grid point by grid point. It was found that severe localization could lead to imbalance, but with large enough radius of influence (decorrelation length) for the measurements, this was not a problem. Hunt et al. use the local analysis method in their ETKF algorithm and developed a local ensemble transform Kalman filter (LETKF) [23].

To eliminate the small background error covariance associated with remote observations, Houtekamer and Mitchell uses a Schur (element-wise) product of a correlation function with local support and the covariance of the background error calculated from the ensemble [14]. That is, the matrix B in Eq. (48) is replaced by $\rho \circ B$, where "$\circ$" represents the element-wise

product and the elements ρ relates to the distance r of the grid point to the observation r as below:

$$\rho(r) = \left(1 + \alpha r + \frac{\alpha^2 r^2}{3}\right) e^{-\alpha r}. \tag{66}$$

Here, α is a scalar parameter. To the best of author's knowledge, this is the first case that the covariance localization is used in EnKF.

Nowadays, a typical covariance localization approach is used to represent prior covariances using an element-wise product of ensemble covariance and a correlation function with compact support [24]. Anderson applied this approach to the Data Assimilation Research Testbed system [25], which has been used for realistic cases.

5. General form of ensemble-based filters for Gaussian models

In proceeding sections, we introduced Kalman-based filters. Originally Kalman filter applies linear model and linear measurement function. Further, EKF and EnKF were developed to address nonlinear models. However, the measurement functions are still assumed to be linear. Eqs. (46) and (47) can directly evaluate nonlinear measurement functions but they were proposed intuitionally and not proven yet. In this section, we will present a general form for nonlinear measurement function and further prove Eqs. (46) and (47) mathematically using the general form.

For generality, we assume the nonlinear model as Eqs. (34) and (35):

$$x_{t+1} = f(x_t) + \eta_t, \tag{67}$$

$$y_t = h(x_t) + \zeta_t, \tag{68}$$

where f and h are nonlinear operators of model and measurement. x is model state and y is the observation. η_t and ζ_t are the model errors and observed errors, respectively, which have variance $var(\eta_t) = <\eta_t, \eta_t^T> = Q, var(\eta_t) = <\zeta_t, \zeta_t^T> = R$. Assuming the estimate of model state x^a at a time step is a linear combination of model forecast x^b and observation y^o, i.e. the filter itself is linear, so

$$x^a = x^b + K\left[y^o - h(x^b)\right] \tag{69}$$

Denoting $\hat{x}^a = x^t - x^a$, $\hat{x}^b = x^t - x^b$, $\hat{y} = y^o - h(x^b)$, we have

http://dx.doi.org/10.5772/64718

$$\hat{x}^a = \hat{x}^b - K\hat{y} \tag{70}$$

$$\begin{aligned} P^a &= E[\hat{x}^a(\hat{x}^a)^T] = E\left[\left(\hat{x}^b - K\hat{y}\right)\left(\hat{x}^b - K\hat{y}\right)^T\right] \\ &= E[\hat{x}^b(\hat{x}^b)^T - \hat{x}^b\hat{y}^T K^T - K\hat{y}(\hat{x}^b)^T + K\hat{y}\hat{y}^T K^T \\ &= P^b - P_{\hat{x}\hat{y}}K^T - KP_{\hat{y}\hat{x}} + KP_{\hat{y}\hat{y}}K^T \end{aligned} \tag{71}$$

The optimal estimate asks the trace of P^a minimum, namely,

$$\frac{\partial[\text{trace}(P^a)]}{\partial K} = -P_{\hat{x}\hat{y}} - P_{\hat{y}\hat{x}} + 2KP_{\hat{y}\hat{y}} = 0, \tag{72}$$

where we invoked the below properties:

$$\frac{\partial\left(\text{trace}\left[XAX^T\right]\right)}{\partial X} = X\left(A + A^T\right) = 2XA, \tag{73}$$

$$\frac{\partial(\text{trace}[XA^T])}{\partial X} = \frac{\partial(\text{trace}[AX^T])}{\partial X} = A^T = A., \tag{74}$$

Thus, we have the optimal estimate filter:

$$x_t^a = x_t^b + K\left[y^o - h\left(x_t^b\right)\right] \tag{75}$$

$$K = P_{\hat{x}\hat{y}}P_{\hat{y}\hat{y}}^{-1} \tag{76}$$

$$P^a = P^b - KP_{\hat{x}\hat{y}}, \tag{77}$$

Eqs. (75)–(77) give a general algorithm for Gaussian nonlinear model and nonlinear measurement function. The first term of Eq. (74) can be interpreted as the cross-covariance $P_{\hat{x}\hat{y}}$ between the state and observation errors, and the remaining expression can be interpreted as the

error covariance $P_{\hat{y}\hat{y}}$ of the difference between model observation and observation itself. Here, $\hat{y}$ is defined as the error between the noisy observation y^o and its prediction $h\left(x^b\right)$.

If the model is linear, obviously,

$$x_{t+1}^b = Mx_t^a + \eta_t,, \tag{78}$$

$$B_{t+1} = MP_t^a M^T + Q. \tag{79}$$

If the measurement function is linear, i.e.

$$\hat{y} = y^o - h\left(x^b\right) - \zeta = y^o - Hx^b - \zeta = Hx^{tr} - Hx^b - \zeta = H\hat{x}^b - \zeta \tag{80}$$

$$P_{\hat{x}\hat{y}} = <\hat{x}^b, \hat{y}^T> = P^b H^T \tag{81}$$

$$P_{\hat{y}\hat{y}} = <\hat{y}, \hat{y}^{\mathrm{T}}> = HP^b H^T + R \tag{82}$$

So, Kalman gain

$$K = P^b H^T (HP^b H^T + R)^{-1} \tag{83}$$

Eq. (83) is identical to Eq. (28). Therefore, Eq. (28), or KF, EKF and EnKF, is a special case of Eq. (76) under the assumption of linear measurement function.

In the standard KF, the state error covariance is updated at each analysis cycle during the measurement update process. Updating the error covariance matrix is important because it represents the change in forecast error covariance when a measurement is performed. The EnKF implementation does not require the covariance update equation because it can directly calculate the updated error covariance matrix from a set of ensemble members. Evensen [17] has derived the analysis of covariance equation that is consistent with the standard KF error covariance to update Eq. (28). But the true representation of the updated error covariance requires a large ensemble size, which is often computationally infeasible.

The general form of the Kalman gain makes use of the reformulated error covariance. In a broad sense, the above algorithm implicitly uses the prior covariance update equation (or the analysis error covariance matrix) to calculate the forecast error covariance. Thus, the above algorithm is fully consistent with the time update and measurement update formulation of the Kalman filter algorithm. On this basis, one can develop a new type of Kalman filter that chooses the ensemble members deterministically in such a way that they can capture the statistical

moments of the nonlinear model accurately. In the next subsection, we will discuss the new type of Kalman filter, called sigma-point Kalman filter, based on the above algorithm.

6. Sigma-point Kalman filters (SPKF)

6.1. Basics of SPKF

EnKF was developed in order to overcome the linearization of nonlinear models. As introduced earlier, the idea behind EnKF is to 'integrate' Fokker-Plank equation using ensemble technique to estimate the forecast error covariance. Theoretically, if the ensemble size is infinite, the estimate approaches the true value. However, in reality, we can only use finite ensemble size, even very small size for many problems, leading to truncation errors. Thus, some concerns exist such as how to wisely generate finite samples for the optimal estimate of prediction error covariance, how much the least ensemble size is for an efficient estimate of error covariance and how much the true error covariance can be taken into account in the EnKF, given an ensemble size. In this section, we will introduce a new ensemble technique for EnKF, which is called sigma-point Kalman filter (SPKF).

The so-called sigma-point approach is based on deterministic sampling of state distribution to calculate the approximate covariance matrices for the standard Kalman filter equations. The family of SPKF algorithms includes the unscented Kalman filter (UKF [26]), the central difference Kalman filter (CDKF [27]) and their square root versions [28]. Another interpretation of the sigma-point approach is that it implicitly performs a statistical linearization of the nonlinear model through a weighted statistical linear regression (WSLR) to calculate the covariance matrices [29]. In SPKF, the model linearization is done through a linear regression between a number of points (called sigma points) drawn from a prior distribution of a random variable rather than through a truncated Taylor series expansion at a single point. It has been found that this linearization is much more accurate than a truncated Taylor series linearization [28]. Eqs. (80)–(82) construct a core of SPKF. A central issue here is how to generate the optimal ensemble members for applying these equations. There are two basic approaches aforementioned, UKF and CDKF. For an L-dimensional dynamical system represented by a set of discretized state-space equations of (67), it has been proven that $2L + 1$ ensemble members, constructed by UKF or CDKF, can precisely estimate the mean and covariance. We ignore the theoretical proof and only outline the UKF scheme as below.

Denote $2L + 1$ sigma points at time k for producing ensemble members by $\chi_k = [\chi_{k,0}, \chi^+_{k,1}, ..., \chi^+_{k,L}, \chi^-_{k,1}, ..., \chi^-_{k,L}]$, which that is defined according to the following expressions:

$$\chi_{k,0} = \bar{X}^a_k \tag{84}$$

$$\chi_{k,i}^{+} = \bar{X}_k^a + [c\sqrt{P_{X,k}^a}]_i \tag{85}$$

$$\chi_{k,i}^{-} = \bar{X}_k^a - [c\sqrt{P_{X,k}^a}]_i \tag{86}$$

where $L = N_x + N_\eta + N_\zeta$ is the sum of the dimensions of model states, model noise and measurement noise. The augmented state vector $X = [x; \eta; \zeta]$ is a L-dimensional vector. $\sqrt{P_{X,k}^a}$ is the covariance of the augmented state vector (analysis) at the previous step. $[\sqrt{P_{X,k}^a}]_i$ is the ith row (column) of the weighted matrix square root of the covariance matrix (L dimension). c is a scale parameter that will be specified later. The key point here is to produce $(2L + 1)$ ensemble members by integrating model with $2L + 1$ initial conditions of Eqs. (84)–(86); by these ensemble members, the filter Eqs. (80)–(82) will be performed.

The procedure is summarized as below:

1. Initially, perturb a small amount, denoted by $\tilde{x}_0$ on initial condition x_0, using Evensen method [17]; and also randomly generate perturbation for q and r, drawn from normal distributions of $N(0, Q)$ and $N(0, R)$. Thus, we can construct the augmented state vector and corresponding covariance ($k = 0$)

$$\bar{X}_0^a = \left[x_0; 0; 0\right]; \tag{87}$$

$$P_0^x = \tilde{x}_0 \tilde{x}_0^T; \tag{88}$$

$$P_{X,0} = \begin{pmatrix} P_0^x & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{pmatrix}. \tag{89}$$

2. From the above formula, we can calculate sigma points using Eqs. (84)–(86). Note that each set of sigma points, denoted by $\chi_{k'}$ has dimension L, e.g. the ith sigma point can be expressed by $\chi_{k,i} = [x_{k,i}; \eta_{k,i}; \zeta_{k,i}]$.

3. Using the $2L + 1$ sigma points to integrate state-space model. For the ith sigma point, we have $x_{k+1,i}^f = f(x_{k,i}, \eta_{k,i})$. When i varies from 1 to $2L + 1$, we produce $2L + 1$ ensemble members, from which analysis mean and covariance will be obtained, which are in turn

http://dx.doi.org/10.5772/64718

used to produce sigma points for next step ($k + 1$), to form a recursive algorithm. Suppose we have $2L + 1$ ensembles, the analysis mean and the covariance are calculated as follows:

$$\bar{x}_{k+1}^{f} = \sum_{i=0}^{2L} w_i^{(m)} x_{k+1,i}^{f} \tag{90}$$

$$(P_{xx}^{f})_{k+1} = \sum_{i=0}^{2L} w_i^{(c)} [x_{k+1,i}^{f} - \bar{x}_{k+1}^{f}][x_{k+1,i}^{f} - \bar{x}_{k+1}^{f}]^T \tag{91}$$

$$y_{k+1,i}^{f} = h\left(x_{k+1,i}^{f}, \zeta_{k+1,i}\right) \tag{92}$$

$$\bar{y}_{k+1}^{f} = \sum_{i=0}^{2L} w_i^{(m)} y_{k+1,i}^{f} \tag{93}$$

$$(P_{yy})_{k+1} = \sum_{i=0}^{2L} w_i^{(c)} [y_{k+1,i}^{f} - \bar{y}_{k+1}^{f}][y_{k+1,i}^{f} - \bar{y}_{k+1}^{f}]^T \tag{94}$$

$$(P_{xy})_{k+1} = \sum_{i=0}^{2L} w_i^{(c)} [x_{k+1,i}^{f} - \bar{x}_{k+1}^{f}][y_{k+1,i}^{f} - \bar{y}_{k+1}^{f}]^T \tag{95}$$

$$K_{k+1} = P_{xy} P_{yy}^{-1}, \tag{96}$$

$$\bar{x}_{k+1}^{a} = \bar{x}_{k+1}^{f} + K_{k+1}\left[y_{k+1} - \bar{y}_{k+1}^{f}\right] \tag{97}$$

$$P_{k+1}^{a} = (P_{xx}^{f})_{k+1} - K_{k+1} P_{yy} K_{k+1}^{T}, \tag{98}$$

where

$$\mathrm{c} = \sqrt{L + \lambda} \tag{99}$$

$$w_0^{(m)} = \frac{\lambda}{L + \lambda} \tag{100}$$

$$w_0^{(c)} = \frac{\lambda}{L+\lambda} + 1 - \alpha^2 + \beta \tag{101}$$

$$w_i^{(m)} = w_i^{(c)} = \frac{1}{2(L+\lambda)}, i = 1,2,...2L \tag{102}$$

$$\lambda = \alpha^2 \left(L + \kappa\right) - L, \tag{103}$$

α and κ are tuning parameters. $0 < \alpha < 1$ and $\kappa \geq 0$. Often κ is chosen 0 as default value and $\beta = 2$.

4. From P_{k+1}^{a}, as well choosing random perturbation for model noise η and observation noise ζ, drawn from Gaussian distribution of $N(0, Q)$ and $N(0, R)$, we calculate sigma points using Eqs. (84)–(86), and repeat Step 2 and Step 3 and so on until the assimilation is completed for the entire period.

6.2. Remarks of SPKF

SPKF was recently introduced into the earth sciences [15, 30]. The main differences between SPKF and EnKF include

i. SPKF chooses the ensemble members deterministically while EnKF uses random perturbation to generate ensemble members;

ii. the number of sigma points is a fixed value as $2L + 1$, while the ensemble size in EnKF is pre-specified;

iii. SPKF uses Eq. (98) to update the error covariance matrix, while EnKF does not update explicitly the error covariance matrix; and

iv. Sigma points are calculated using Eqs. (84)–(86) every time when the observation is available, while the ensemble members in EnKF only perturbed in the initial time. Recent application of SPKF on a realistic oceanic model indicates that the SPKF is better than the EnKF in the similar level of computational cost [31].

In SPKF, the number of sigma points is $2L + 1$, here L is the dimension of the augmented state vector $X = [x; \eta; \zeta]$, i.e. $L = N_x + N_\eta + N_\zeta$ is the sum of model state, model noise and observation noise. Usually, L is the order 10^3–10^4, so the computational expense is a huge challenge in SPKF for realistic problems. A solution is to use the truncated singular-value decomposition (TSVD) to reduce the sigma points. As seen from Eqs. (84)–(86), the $P_{X,k}^{a}$ is a $L * L$ matrix, thus the dimension of $P_{X,k}^{a}$ determines the ensemble size. Suppose that $P_{X,k}^{a}$ can be expressed as

http://dx.doi.org/10.5772/64718

$$P_{X,k}^{a} = E_{X,k}^{a} \Sigma_k (E_{X,k}^{a})^T \tag{104}$$

where $\Sigma_k = \text{diag}(\sigma_k^1, \sigma_k^2, ..., \sigma_k^L)$ is a diagonal matrix of eigenvalues that are sorted in descending order, i.e. $\sigma_k^1 \geq \sigma_k^2 \geq ... \geq \sigma_k^L$, and $E_{X,k}^{a} = [e_{X,k,1}^{a}, e_{X,k,2}^{a}, ..., e_{X,k,L}^{a}]$. Truncating the first m modes, so we can write the sigma points (84)–(86) as below:

$$\chi_{k,0} = \bar{X}_k^a \tag{105}$$

$$\chi_{k,i}^{+} = \bar{X}_k^a + c\sqrt{\sigma_k^i} e_{X,k,i}^{a} \tag{106}$$

$$\chi_{k,i}^{-} = \bar{X}_k^a - c\sqrt{\sigma_k^i} e_{X,k,i}^{a} \tag{107}$$

$i = 1, 2, ..., m$. Thus, the ensemble size becomes $2 * m + 1$, where $m << L$. Some fast SVD algorithms can be used here, such as Lanczos and block Lanczos [32]. The application of the truncated SVD was also found in [33, 34].

Further simplifying $P_{X,k}^{a}$ based on its definition (or Cholesky decomposition), i.e. $P_{X,k}^{a} = A_{X,k}^{a} * (A_{X,k}^{a})^T$, where $A_{X,k}^{a}$ is the data that has subtracted the ensemble mean. Thus, Eqs.(82)–(84) can be written as follows:

$$\chi_{k,0} = \bar{X}_k^a \tag{108}$$

$$\chi_{k,i}^{+} = \bar{X}_k^a + [cA_{X,k}^{a}]_i \tag{109}$$

$$\chi_{k,i}^{-} = \bar{X}_k^a - [cA_{X,k}^{a}]_i \tag{110}$$

where $[cA_{X,k}^{a}]_i = [x_k^a; \eta_k; \zeta_k]_i, i = 1, 2, ..., L$, $(x_k^a)_i = (x_k^f)_i + K_k[y_k - y_k^f]$. Eqs.(109) and (110) transfer the covariance matrix $P_{X,k}^{a}$ to data matrix $A_{X,k}^{a}$ in constructing sigma points. The largest advantage is to avoid explicit expression of $P_{X,k}^{a}$, which could be a very large matrix beyond memory of current computers. However, Eqs.(109) and (110) cannot reduce the ensemble size $(2L + 1)$. A solution is to decompose, such as principal component analysis, as

used in [14]. Further discussions on optimal construction of sigma points should be conducted for a realistic application of SPKF.

Again, we look at sigma-point generation, i.e. Eqs. (106) and (107) or (109) and (110). As we defined, an augmented matrix is applied here $[x;\eta;\zeta]$. Without losing the generality, rewrite them as below:

$$\begin{bmatrix} x_{k,0} \\ \eta_{k,0} \\ \zeta_{k,0} \end{bmatrix} = \begin{bmatrix} \overline{x}_{k,0} \\ 0 \\ 0 \end{bmatrix} \tag{111}$$

$$\begin{bmatrix} x_{k,i} \\ \eta_{k,i} \\ \zeta_{k,i} \end{bmatrix} = \begin{bmatrix} \overline{x}_{k,0} \\ 0 \\ 0 \end{bmatrix} + c \begin{bmatrix} x^a_{k,i} \\ \eta_{k,i} \\ \zeta_{k,i} \end{bmatrix} \tag{112}$$

Similarly, we can write Eq. (107) or (110) using individual variables. From Eqs. (111) and (112), we can draw

- Noise and model state analyses in constructing sigma points at k step are independent. It should be noted that x^a_k is from Eq. (97) and noise are draw from a Gaussian distribution. If we assume that noise is taken randomly each time, x^a_k is only relevant to noise that is drawn at time step k, and independent with model noise and observation noise drawn for analysis of the time step $k+1$, thus, $P_{X,k}$ is a diagonal block matrix, i.e.

$$P_{X,k} = \begin{pmatrix} P^x_k & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{pmatrix} \tag{113}$$

- There are no update equations for noise, so they are randomly taken from Gaussian distribution, i.e. the index i in η_i and ζ_i actually does not have meaning. Thus, it should be a reasonable assumption that the η_i and ζ_i, used for constructing sigma points at time step $k+1$, are not related to $P_{X,k}$ (time step of k), as argued above. Thus, Eq. (108) always holds unless the noise is designed considering the temporal coherence such as red noise in time.
- Based on the above, the actual ensemble size is $2N_x+1$, and not $2L+1$. This is because neither model noise nor observation noise can produce ensemble alone. Model errors η_i and $x^f_{k,i}$ must be joined together to produce ensemble members with N_x. Let us see this in details:

at the initial time, initial perturbation on model states plus drawn noise for model errors and measurement errors are with mean and variance as follows:

$$\bar{X}_0^a = \left[x_0; 0; 0\right], \quad P_{X,0} = \begin{pmatrix} P_0^x & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{pmatrix} \tag{114}$$

Theoretically, there are $2(N_x + N_\eta + N_\zeta) + 1$ ensembles, denoted by the ith column of $P_{X,0}$ ($i = 1, ..., N_x; Nx + 1, ..., N_x + N_\eta; N_x + N_\eta + 1, ..., N_x + N_\eta + N_\zeta$) and formula (84)–(86). However, at the ith column, the elements of the row, indicating the model inputs (x, η, ζ), only have the non-zero values of N_x. Obviously, the sigma points of zero-values makes the update equation $\chi_{k+1,i} = f(\chi_{k,i})$ invalid, thus, the actual ensemble size is $2N_x + 1$.

When truncation technique is applied to reduce the ensemble size, the ensemble spread might be shrunk due to relatively small ensemble size. Like EnKF, an inflation approach of SPKF might be helpful. It is interested in developing such a scheme for SPKF. Also, we can localize SPKF, like localized EnKF, to solve memory and computation issues.

All of the remarks of SPKF are from the authors' thinking and understanding. It is interesting to further test and validate these ideas and properties using simple models.

7. Beyond Kalman filters: particle filter and its derivatives

7.1. Standard particle filter

We have introduced the Kalman filter (KF), extended Kalman filter (EKF), ensemble Kalman filter (EnKF) and sigma-point Kalman filter (SPKF) in previous sections. All of those filters belong to the sequential data assimilation method, i.e. observation data is assimilated into the model system as soon as it is available. The Bayesian estimation theory provides a general framework of the sequential data assimilation methods. If we assume the state-space model is given by Eqs. (34) and (35), the analysis step of a Bayesian-based assimilation method is deduced by Bayes' theorem:

$$p\left(x_t|y_t\right) = \frac{p\left(y_t|x_t\right)p\left(x_t\right)}{p\left(y_t\right)}, \tag{115}$$

where $p(y_t)$ plays as a normalization factor.

Recalling Section 2.3, Eq. (12) actually assumes that the prior probability density function $p(x_t)$ and the likelihood function $p(y_t|x_t)$ are Gaussian distribution functions, and thus the posterior

probability density function $p(x_t|y_t)$ is also a Gaussian. Based on the Gaussian assumption, the cost function of 3D-Var (i.e. Eq. (6)) can be derived, and it is equivalent to the Kalman filter Eqs. (27)–(29). All the Kalman-based filters (e.g. EKF, EnKF, EnSRF, SPKF, etc.) contain the inherent Gaussian assumption, and they are derived and validated for Gaussian systems in theory. However, this Gaussian assumption is often not applicable for nonlinear systems. Even for an initial Gaussian error, it often becomes non-Gaussian while propagating forward with nonlinear models.

The particle filter (PF) is a sequential data assimilation method that is able to deal with the nonlinear and non-Gaussian state estimation problem. Like EnKF, PF also uses an ensemble, but it is used to approximately estimate the full probability density function rather than only the error covariance B. An ensemble member is also referred to as a particle in PF literatures. Suppose the prior probability density is the sum of Dirac delta functions

$$p(x_t) = \sum_{i=1}^{N} \delta(x_t - x_t^i) \tag{116}$$

where $\{x_t^i, i = 1, 2, ..., N\}$ are particles drawn from $p(x_t)$. The posterior probability density is derived by applying the Bayes' theorem directly, that is

$$p(x_t|y_t) \propto p(y_t|x_t)\, p(x_t) = \sum_{i=1}^{N} w_{t,i} \delta(x_t - x_t^i) \tag{117}$$

in which $w_{t,i} \propto p(y_t|x_t^i)$, and a normalization step, is required to make $\{w_{t,i}, i = 1, 2, ..., N\}$ sum up to 1. If we assume the likelihood function is Gaussian, $w_{t,i}$ can be computed by

$$p(y_t \mid x_t^i) = \frac{1}{\sqrt{2\pi R}} \exp\{-1/2[y_t - h(x_t^i)]R^{-1}[y_t - h(x_t^i)]^T\}. \tag{118}$$

Or else we can use any specified probability density function of $p(y_t|x_t)$ to compute the likelihood.

With the posterior probability density function $p(x_t|y_t)$, the analysis value and covariance can be computed by

$$\bar{x}_t = \int x * p(x \mid y_t) dx = \sum_{i=1}^{N} w_{t,i} x_t^i \tag{119}$$

$$\mathrm{var}(x_t) = \int x^2 * p(x \mid y_t) dx - \bar{x}_t^2 = \sum_{i=1}^{N} w_{t,i} (x_t^i)^2 - \bar{x}_t^2 \tag{120}$$

and higher-order moments of the posterior state can also be estimated.

Before stepping forward to next stage, a resampling step is required to make each particle with uniform weight. A typical resampling strategy is the sequential importance resampling (SIR) that removes particles with very small weights and duplicates those with large weights. A detailed algorithm of SIR can be found in [35]. The resampling algorithm gives the indices and number of copies of those particles that should be duplicated, i.e. computes $s_1, s_2, ..., s_N$ according to the weights, where each $s_i \in 1, 2, ..., N$. And then $\left\{x_t^{s_i}, i = 1, 2, ..., N\right\}$ are regarded as new particles.

In summary, the algorithm of standard particle filter is given below:

1. generate the initial ensemble $\left\{x_0^i, i = 1, 2, ..., N\right\}$ as EnKF does;
2. integrate the model until the observation is available;
3. use Eq. (118) to compute the weight for each particle, and normalize them;
4. use Eq. (119) to obtain the analysis and Eq. (120) to obtain the covariance if necessary;
5. apply the resampling algorithm to derive the resampling indices, and derive the new ensemble $\left\{x_t^{s_i}, i = 1, 2, ..., N\right\}$; and
6. repeat Steps 2–5 until the end of assimilation period.

The standard particle filter [36] is also known as the bootstrap particle filter or SIR particle filter.

7.2. Variants of PF

The particle filter is a highly promising technique because it does not invoke any Gaussian assumptions. It has been widely used and studied in many other fields. The PF estimates the full probability density function of the forecasted state based on an ensemble of states with different weights. However, the PF suffers from the problem of filter degeneracy, i.e. the procedure collapses to a very small number of highly weighted particles among a horde of almost useless particles carrying a tiny proportion of the probability mass. Even if resampling techniques are used, the degeneracy cannot be completely avoided with limited ensemble size. The number of particles must grow substantially with the dimension of the system to avoid degeneracy [37, 38], a requirement that is apparently too costly for large models such as GCMs. Various efforts have been made to resolve this issue, as documented in an excellent overview [39].

Several strategies are often employed to address the problem of filter degeneracy in applications of the particle filter. For example, Papadakis et al. proposed a weighted ensemble Kalman filter (WEnKF) [40] that uses an ensemble-based Kalman filter as the proposal density from which the particles are drawn. Van Leeuwen et al. developed a fully nonlinear particle filter

by exploiting the freedom of the proposal transition density, which ensures not only that all particles ultimately occupy high-probability regions of state-space but also that most of the particles have similar weights [41]. The implicit particle filter uses gradient descent minimization combined with random maps to find the region of high probability, avoiding the calculation of Hessians [42]. Luo et al. have proposed an efficient particle filter that uses residual nudging to prevent the residual norm of the state estimates from exceeding a prespecified threshold [43]. These particle filters were very recently proposed and have attracted broad attention in the community of atmos./ocean. data assimilation. Below, we will briefly introduce the equivalent weights particle filter (EWPF) by Van Leeuwen [39, 41].

The equivalent weights particle filter is a fully nonlinear data assimilation method that works in a two-stage process. It uses the proposal density to ensure that the particles have almost equivalent weights, by which the filter degeneracy can be avoided.

In the standard PF, the particles at time step t are propagated by the original model, i.e. $x_{t+1}^i = f(x_t^i) + \eta_t$, which implies that the particles at time step $t+1$ are drawn from the transition density $p(x_{t+1}|x_t)$. In that case, the weight of each x_{t+1}^i varies greatly and filter degeneracy is very likely to happen.

In EWPF, another transition density, call the proposal density, is introduced. The proposal density depends on the future observation y_{t+1} and all previous particles $\left\{x_t^i, i = 1, 2, ..., N\right\}$. With the help of proposal density, the particle x_t^i is propagated using a different model

$$x_{t+1}^i = g\left(x_t^i, y_{t+1}\right) + \eta_t. \tag{121}$$

The model g can be anything, for instance, one can add a relaxation term and change random forcing:

$$x_{k+1}^i = f\left(x_k^i\right) + \eta_k^i + A\left(y_{t+1} - H\left(x_k^i\right)\right), \quad k = 1,\ldots, p\left(k\right) \tag{122}$$

where $p(k)$ is a function of the time between observations, and each k implies each model step without observation. A is a relaxation term that will 'drag' the particle towards future observation. In [44], it is given by

$$A = p\left(k\right)QH^T R^{-1}, \tag{123}$$

where the matrices Q and R correspond to the model error covariance and observation error covariance, respectively.

The second stage of EWPF involves updating each particle at the observation time $t + 1$ via the formula

$$x_{t+1}^{i} = f(x_t^i) + \alpha_i QH^T (HQH^T + R)^{-1} (y_{t+1} - H(f(x_t^i))) + \eta_t^i \tag{124}$$

where α_i are scalers computed so as to make the weights of the particles equal. Using the expression for weights and setting all weights equal to a target weight (e.g. $1/N$)

$$w_i = p\left(y_{t+1} | x_{t+1}^i(\alpha_i)\right) = w_{\text{target}} \tag{125}$$

α_i can be solved by numerical methods.

Eqs. (122)–(125) show an example of how to construct the proposal model g in(121)), it can also be done by running 4D-var on each particle (implicit particle filter), or using the EnKF as proposal density. Those methods refer to Morzfeld et al. [42] and Papadakis et al. [40].

7.3. Remarks of PF

7.3.1. Combined method of EnKF and PF

The ensemble Kalman particle filter (EnKPF) is a combination of the EnKF and the SIR particle filter. It was recently introduced to address non-Gaussian features in data assimilation for highly nonlinear systems, by providing a continuous interpolation between the EnKF and SIR-PF analysis schemes [45].

As stated above, both EnKF and PF methods are based on the Bayesian estimation theory, but they approximate the probability density function of the state in different ways. The EnKF only approximates the mean and covariance of the state through a series of equally weighted ensemble members. And the particle filter considers the weights of the ensemble members according to the likelihoods. The EnKF contains the Gaussian assumption but requires relatively small ensemble size to prevent filter degeneracy, which is in contrast with the PF.

The EnKPF takes advantage of both methods by combining the analysis schemes of the EnKF and the SIR-PF using a controllable index (i.e. tuning parameter). In contrast with both the EnKF and the SIR-PF, the analysis scheme of the EnKPF not only updates the ensemble members but also considers the weights.

Assume that the forecast ensemble $\left\{x_i^f, i = 1, 2, ..., N\right\}$ and the observation data y are available, and that the forecast covariance P^f can be calculated using the ensemble, the analysis scheme of EnKPF is given below.

1. Choose $\gamma \in [0, 1]$ and apply the EnKF that is based on the inflated observation error covariance R/γ as follows:

$$K_1(\gamma) = P^f H^T (HP^f H^T + R/\gamma)^{-1} = \gamma P^f H^T (\gamma HP^f H^T + R)^{-1} \tag{126}$$

$$v_i = x_i^f + K_1(\gamma)(y - Hx_i^f) \tag{127}$$

$$Q = \frac{1}{\gamma} K_1(\gamma) R K_1(\gamma)^T \tag{128}$$

2. Compute the weights w_i for each updated member v_i as follows:

$$w_i = \phi\left(y; Hv_i, \frac{R}{1-\gamma} + HQH^T\right) \tag{129}$$

and normalize the weights by $\hat{w}_i = w_i / \sum_{i=1}^{N} w_{i'}$ in which φ is the probability density function of a Gaussian.

3. Calculate the resampling index $s(i)$ for each member v_i according to $\hat{w}_i$ using the SIR algorithm, then set

$$x_i^u = v_{s(i)} + K_1(\gamma)\frac{\epsilon_{1,i}}{\sqrt{\gamma}} \tag{130}$$

where $\epsilon_{1,i}$ is a random observation error drawn from the Gaussian $N(0, R)$.

4. Compute $K_2(1-\gamma) = (1-\gamma)QH^T[(1-\gamma)HQH^T + R]^{-1}$, and generate $\epsilon_{2,i}$ from $N(0, R)$ and EnKF with the inflated observation error again as follows:

$$x_i^a = x_i^u + K_2(1-\gamma)\left[y + \frac{\epsilon_{2,i}}{\sqrt{1-\gamma}} - Hx_i^u\right] \tag{131}$$

γ can be determined recursively to match the optimal performance of EnKPF. More details of EnKPF can be found in [45, 46].

7.3.2. Localization in PF

Previous sections have introduced the localization technique in EnKF, which greatly improves the performance of EnKF in high-dimensional models. The advantages of localization motivate the search for a localization procedure in particle filtering.

http://dx.doi.org/10.5772/64718

Van Leeuwen had a deep discussion on this topic [39]. He argued that *one can calculate the weights locally, but it is not easy for resampling. In the resampling step low-weight particles are abandoned and high-weight particles are duplicated. However, with local weights, different particles are selected in different parts of the domain. The problem is that we have to have continuous (in space) model fields to propagate forward in time with the model. Just constructing a new particle that consists of one particle in one part of the model domain and another particle in another domain will lead to problems at the boundary between these two*.

The problem of spatial discontinuity makes the localization in particle filter not feasible currently. Most of the advanced particle filters (e.g. EWPFand implicit particle filter) are using the idea of global weight, i.e. the weight for each member is a scalar.

However, there are still some attempts on the localization in particle filter. For example, Poterjoy developed the localized particle filter (LPF) that updates particles locally using ideas borrowed from EnKF [47]. The paper has demonstrated some advantages of the new filter over EnKF, especially when the observation networks consist of densely spaced measurements that relate nonlinearly to the model state. This is a very interesting work about the particle filter, it also has a potential to work with large atmos./ocean. data assimilation systems.

8. Remarks and conclusions

Data assimilation is the process by which observations of the actual system are incorporated into a numerical model to optimally estimate the system states. In this chapter, we introduced several ensemble-based data assimilation methods that are widely used in the earth sciences. One can read it as an introduction to ensemble-based data assimilation methods, but also can view it as a brief review of the application of these ensemble-based assimilation methods on the earth sciences. It is author's effort to write such a 'review' chapter with introductory language, making it more readable. As found in the chapter, many discussions, derivations and analyses are actually very thoughtful, not only introducing these methods, but also deepening the understanding to them. This is emphasized by the analysis of the rationale behind each method, including: i). the principle for deriving the algorithm; ii) basic assumptions of each method; iii). the connection and relation of different methods (e.g., EKF and EnKF, EnKF and SPKF etc.); iv). the advantages and deficiencies of each method. Especially we put rather weights to discuss potential concerns, challenges and possible solutions when these methods are applied to high-dimensional systems in the earth sciences. This chapter can be a "textbook" for the beginners to learn these data assimilation algorithms, and also a good reference for researchers for better understanding and applying these methods.

Acknowledgements

This work was supported by the NSERC (Natural Sciences and Engineering Research Council of Canada) Discovery Grant, the National Science Foundation of China (41276029, 41321004,

41530961,91528304), the National Programme on Global Change and Air-Sea Interaction (GASI-IPOVAI-06)and the National Basic Research Program (2013CB430302).

Author details

Youmin Tang[1,2*], Zheqi Shen[2] and Yanqiu Gao[2]

*Address all correspondence to: ytang@unbc.ca

1 Environmental Science and Engineering, University of Northern British Columbia, Prince George, British Columbia, Canada

2 State Key Laboratory of Satellite Ocean Environment Dynamics, Second Institute of Oceanography, State Oceanic Administration, Hangzhou, China

References

[1] Talagrand O. Assimilation of observations, an introduction. Journal of Meteorological Society of Japan Series 2. 1997;75:81–99.

[2] Kalnay E. Atmospheric modeling, data assimilation, and predictability. Cambridge University Press. New York. 2003.

[3] Le Dimet F. X., Talagrand O. Variational algorithms for analysis and assimilation of meteorological observations: Theoretical Aspects. Tellus A. 1986;38(2):97–110.

[4] Courtier P., Andersson E., Heckley W., Vasiljevic D., Hamrud M., Hollingsworth A. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I:Formulation. Quarterly Journal of the Royal Meteorological Society. 1998;124(550): 1783–1807.

[5] Kalman R. E. A new approach to linear filtering and prediction problems. Journal of Basic Engineering. 1960;82(1):35–45.

[6] Rabier F., Jarvinen H., Klinker E., Mahfouf J. F., Simmons A. The ECMWF operational implementation of four-dimensional variational assimilation. I: Experimental results with simplified physics. Quarterly Journal of the Royal Meteorological Society. 2000;126(564):1143–1170.

[7] Mahfouf J. F., Rabier F. The ECMWF operational implementation of four-dimensional variational assimilation. II: Experimental results with improved physics. Quarterly Journal of the Royal Meteorological Society. 2000;126(564):1171–1190.

[8] Klinker E., Rabier F., Kelly G., Mahfouf J. F. The ECMWF operational implementation of four-dimensional variational assimilation. III: Experimental results and diagnostics

with operational configuration. Quarterly Journal of the Royal Meteorological Society. 2000;126(564):1191–1215.

[9] Houtekamer P. L., Mitchell H. L., Pellerin G., Buehner M., Charron M., Spacek L., et al. Atmospheric data assimilation with an ensemble Kalman filter: Results with real observations. Monthly Weather Review. 2005;133(3):604–620.

[10] Evensen G. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. Journal of Geophysical Research: Oceans (1978–2012). 1994;99(C5):10143–10162.

[11] Houtekamer P. L., Mitchell H. L. Data assimilation using an ensemble Kalman filter technique. Monthly Weather Review. 1998;126(3):796–811.

[12] Evensen G. Advanced data assimilation for strongly nonlinear dynamics. Monthly Weather Review. 1997;125(6):1342–1354.

[13] Keppenne C. L. Data assimilation into a primitive-equation model with a parallel ensemble Kalman filter. Monthly Weather Review. 2000;128:1971–1981.

[14] Houtekamer P. L., Mitchell H. L. A sequential ensemble Kalman filter for atmospheric data assimilation. Monthly Weather Review. 2001;129(1):123–137.

[15] Ambadan J. T., Tang Y. Sigma-point Kalman filter data assimilation methods for strongly. Journal of the Atmospheric Sciences. 2009;66(2):261–285.

[16] Tang Y., Ambandan J., Chen D. Nonlinear measurement function in the ensemble Kalman filter. Advances in Atmospheric Sciences. 2014;31(3):551–558.

[17] Evensen G. The ensemble Kalman filter: Theoretical formulation and practical implementation. Ocean Dynamics. 2003;53(4):343–367.

[18] Burgers G., van Leeuwen P. J., Evensen G. Analysis scheme in the ensemble Kalman filter. Monthly Weather Review. 1998;126(6):1719–1724.

[19] Whitaker J. S., Hamill T. M. Ensemble data assimilation without perturbed observations. Monthly Weather Review. 2002;130(7):1913–1924.

[20] Tippett M. K., Anderson J. L., Bishop C. H., Hamill T. M., Whitaker J. S. Ensemble square root filters. Monthly Weather Review. 2003;131(7):1485–1490.

[21] Bishop C. H., Etherton B. J., Majumdar S. J. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. Monthly Weather Review. 2001;129(3):420–436.

[22] Anderson J. L. An ensemble adjustment Kalman filter for data assimilation. Monthly Weather Review. 2001;129(12):2884–2903.

[23] Hunt B. R., Kostelich E. J., Szunyogh I. Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. Physica D: Nonlinear Phenomena. 2007;230(1):112–126.

[24] Gaspari G., Cohn, S. Construction of correlation functions in two and three dimensions. Quarterly Journal of the Royal Meteorological Society. 1999;125(554):723–757.

[25] Anderson J. L. Ensemble Kalman filters for large geophysical applications. IEEE Control Systems. 2009;29(3):66–82.

[26] Julier S. J., Uhlmann J. K., Durrant-Whyte H. F. A new approach for filtering nonlinear systems. Proceedings of American Control Conference. 1995;3:1628–1632.

[27] Ito K., Xiong K. Gaussian filters for nonlinear filtering problems. IEEE Transactions on Automatic Control. 2000;45(5):910–927.

[28] van Der Merwe R., Wan E. Efficient derivative-free Kalman filters for online learning. In: Proceedings of ESANN. 2001.

[29] Gelb A. Applied optimal estimation. MIT Press. Cambridge. 1974.

[30] Luo X., Morez I. M. Ensemble Kalman filter with the unscented transform. Physica D: Nonlinear Phenomena. 2009;238(5):549–562.

[31] Tang Y., Deng Z., Manoj K. K., Chen D. A practical scheme of the sigma-point Kalman filter for high-dimensional systems. Journal of Advances in Modeling Earth Systems. 2014;6:21–37.

[32] Golub G. H., van Loan C. F. Matrix computations. JHU Press. Maryland. 2012.

[33] Hansen B. Atmospheric data assimilation with an ensemble Kalman filter: Results with real observations. Monthly Weather Review.2005;133(3):604–620.

[34] Ehrendorfer M., Tribbia J. J. Optimal prediction of forecast error covariance through singular vectors. Journal of the Atmospheric Sciences. 1997;52(2):286–313.

[35] Arulampalam M. S., Maskell S., Gordon N., Clapp T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing. 2002;50(2):174–188.

[36] Gordon N. J., Salmond D. J., Smith A. F. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings F (Radar and Signal Processing). 1993;140(2):107–113.

[37] Bengtsson T., Snyder C., Nychka D. Toward a nonlinear ensemble filter for high-dimensional systems. Journal of Geophysical Research. 2003;108(D24):8775.

[38] Snyder C., Bengtsson T., Bickel P., Anderson J. Obstacles to high-dimensional particle filtering. Monthly Weather Review. 2008;136(12):4629–4640.

[39] van Leeuwen P. J. Particle filtering in geophysical systems. Monthly Weather Review. 2009;137(12):4089–4114.

[40] Papadakis N., Mémin E., Cuzol A., Gengembre N. Data assimilation with the weighted ensemble Kalman filter. Tellus A. 2010;62(5):673–697.

[41] van Leeuwen P. J. Efficient nonlinear data-assimilation in geophysical fluid dynamics. Computers and Fluids. 2011;46(1):52–58.

[42] Morzfeld M., Chorin A. J. Implicit particle filtering for models with partial noise, and an application to geomagnetic data assimilation. Nonlinear Processes in Geophysics. 2012;19(3):365–382.

[43] Luo X., Hoteit I. Efficient particle filtering through residual nudging. Quarterly Journal of the Royal Meteorological Society. 2014;140(679):557–572.

[44] Browne P. A., van Leeuwen P. J. Twin experiments with the equivalent weights particle filter and HADCM3. Quarterly Journal of the Royal Meteorological Society. 2015;141(693):3399–3414.

[45] Frei M., Kunsch H. R. Bridging the ensemble Kalman and particle filters. Biometrika. 2013;100(4):781–800.

[46] Shen Z., Tang Y. A modified ensemble Kalman particle filter for non-Gaussian systems with nonlinear measurement functions. Journal of Advances in Modeling Earth Systems. 2015;7(1):50–66.

[47] Poterjoy J. A localized particle filter for high-dimensional nonlinear systems. Monthly Weather Review. 2016;144(1):59–76.

Section 2

Control and Applications of Nonlinear Dynamical Systems

Chapter 8

Conditions for Optimality of Singular Controls in Dynamic Systems with Retarded Control

Misir J. Mardanov and Telman K. Melikov

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64225

Abstract

In this chapter, we consider an optimal control problem with retarded control and study a larger class of singular (in the classical sense) controls. For the optimality of singular controls, the various necessary conditions in the recurrent forms are obtained. These conditions contain also the analogs of Kelly, Koppa-Mayer, Gabasov, and equality-type conditions. While proving the main results, the Legendre polynomials are used as variations of control.

Keywords: singular control, optimal control, variation transform method, Legendre polynomial, necessary optimality conditions

1. Introduction

As is known, optimal control problems described by the dynamical systems with retarded control are attracting the attention of many specialists, and the results obtained in this field deal mainly with the first-order necessary optimality conditions [1–8, etc.]. However, theory of singular controls for systems with retarded control has not been studied enough yet [9, 10]. One of the main reasons here is that the methods proposed and developed for ordinary systems (for systems without retardation) in [11–18] are not directly applicable to the singular controls in dynamical systems with aftereffect (see [9, 14–19]). Therefore, to study optimal control problems in the systems with retarded control is of special theoretical interest. Besides, such problems have practical significance as well, because mathematical modelling for some problems of organization of the economic plan and production leads to the problems with retarded control (see, e.g., [20]).

As is known, the concept of singular control was first introduced to the theory of optimal processes by Rozenoer [22] in 1959. First results on the necessary optimality conditions for singular controls have been obtained by Kelley [12] in the case of open set U, and by Gabasov [11] in the case of arbitrary (in particular, closed) set U, where U is a set of values of admissible controls. Afterward, Kelley and Gabasov's conditions as well as the methods for treating singular controls proposed in [11, 13] have been significantly generalized in [10, 14–19, 23–41, etc.] to the cases of (1) controls with higher-order degeneration, (2) multidimensional controls, and (3) various classes of control systems. Considering all these cases, the methods in [11, 13] have been generalized in [17, 37] and for optimality of singular controls, necessary conditions in the form of recurrence sequences are obtained for dynamical systems with delayed in state. Similar results for the problem of dynamic systems with retarded control have been obtained in [10] only for singular controls with full degree of degeneration. Below, by considering a larger class of singular controls, proposing a modified version of the variations transform method [13] and matrix impulse method [11], we generalize all results of [10]. While treating the optimality of singular (in the classical sense) controls, we use the Legendre [[42], p. 413] polynomials as variations of control because such an approach is more convenient.

1. Problem statement. Consider the following optimal control problem with retarded control:

$$S(u) = \varphi\left(x(t_1)\right) \to \min_u \tag{1.1}$$

$$\dot{x}(t) = f\left(x(t), u(t), u(t-h), t\right), \quad t \in I := [t_0, t_1], \; x(t_0) = x_0, \tag{1.2}$$

$$u(t) = w(t), \; t \in I_0 := [t_0 - h, t_0), \; u(t) \in U \subset R^r, \; t \in I. \tag{1.3}$$

Here, U is an open set in r -dimensional Euclidean space R^r, $R^1 =: R := (-\infty, +\infty)$, $x \in R^n$ is an n-vector with phase coordinates, $u \in U$ is an r-vector of control actions, $h = \text{const} > 0$, x_0, t_0, t_1 are fixed points with $t_1 > t_0 + h$; $\varphi(x)\colon R^n \to R$, $f(x,u,v,t)\colon R^n \times R^r \times R^r \times R \to R^n$, $w(\cdot) \in \tilde{C}^+\left([t_0 - h, t_0], R^r\right)$ are the given functions, where $\tilde{C}^+\left([t_0 - h, t_0], R^r\right)$ is a class of piecewise continuous (continuous from the right at discontinuity points and continuous from the left at the point t_0) vector functions $w(t)\colon [t_0 - h, t_0] \to R^r$.

The function $u(\cdot)$ is said to be an admissible control if it belongs to $\tilde{C}^+\left(I_1, R^r\right)$ and satisfies the condition (1.3), where $I_1 := I_0 \cup I = [t_0 - h, t_1]$.

Note that if the function $f(\cdot)$ and its partial derivative $f_x(\cdot)$ are continuous on $R^n \times R^r \times R^r \times R$, then, by using the method of successive approximations as in [21] it is easy to show that every admissible control $u(\cdot)$ generates a unique absolutely continuous solution

http://dx.doi.org/10.5772/64225

$x(\cdot)$ of the system (1.2), (1.3) where this solution will be assumed as defined everywhere on I.

If the admissible control $u^0(t),\ t \in I_1$ is a solution of the problem (1.1)–(1.3), we will call it an optimal control, while the corresponding trajectory $x^0(t),\ t \in I$ of the system (1.2)–(1.3) will be called an optimal trajectory. The pair $\left(u^0(\cdot), x^0(\cdot)\right)$ will be called an optimal process.

While studying the problem (1.1)–(1.3), we will also use the following assumptions:

(A1) let the functional $\varphi(x)\colon R^n \to R$ be twice continuously differentiable in the space R^n;

(A2) let the function $f(\cdot)$ and its partial derivatives $f_z(\cdot)$, $f_{zz}(\cdot)$ be continuous in the space $R^n \times R^r \times R^r \times R$, where $z = (x, u, v)$;

(A3) let the function $f(\cdot)$ be three times continuously differentiable in the totality of its arguments in the space $R^n \times R^r \times R^r \times R$;

(A4) let the inclusions $\dot{w}(\cdot) \in \tilde{C}\left(\left[t_0 - h, t_0\right], R^r\right)$ and $\dot{u}^0(\cdot) \in \tilde{C}\left(I_1, R^r\right)$ hold for the derivatives $\dot{w}(\cdot)$ and $\dot{u}^0(\cdot)$, where $\tilde{C}\left([a, b], R^r\right)$ is a class of piecewise continuous (continuous from the right and left at the points a and b, respectively) vector functions $c(t)\colon [a, b] \to R^r$;

(A5) let the function $f(\cdot)$ be sufficiently smooth in the totality of its arguments in the space $R^n \times R^r \times R^r \times R$;

(A6) let the initial function $w(\cdot) \in \tilde{C}^+\left(\left[t_0 - h, t_0\right], R^r\right)$ and admissible control $u^0(\cdot)$ be sufficiently piecewise smooth, that is, $\frac{d^m}{dt^m} w(t) \in \tilde{C}\left(\left[t_0 - h, t_0\right], R^r\right)$ and $\frac{d^m}{dt^m} u^0(t) \in \tilde{C}\left(I_1, R^r\right),\ m = 1, 2, \ldots.$

Especially note that more precise assumptions on the analytic properties of $\varphi(\cdot), f(\cdot), u(\cdot), w(\cdot)$ will directly follow from the representation of optimality criteria obtained below.

2. The second variation of the objective functional and the definition of a singular (in the classical sense) control

Let assumptions (A1) and (A2) be fulfilled, and $\left(u^0(\cdot), x^0(\cdot)\right)$ be some admissible process. If the process $\left(u^0(\cdot), x^0(\cdot)\right)$ is optimal, then, by using the known technique (see, e.g., [27, p. 51]), it is easy to get

$$\delta^1 S\left(u^0;\delta u(\cdot)\right)=0,\delta^2 S\left(u^0;\delta u(\cdot)\right)\geq 0,\ \forall \delta u(\cdot)\in \tilde{C}^{+}\left(I_1,R^r\right),\delta u(t)=0,\ t\in I_0. \tag{2.1}$$

Here

$$\begin{aligned}\delta^1 S\left(u^0;\delta u(\cdot)\right) &:= -\int_{t_0}^{t_1}\left[H_u^T(t)\delta u(t)+H_\upsilon^T(t)\delta u(t-h)\right]dt,\\ \delta u(\cdot)&\in \tilde{C}^{+}\left(I_1,R^r\right),\delta u(t)=0,\ t\in I_0,\end{aligned} \tag{2.2}$$

$$\begin{aligned}\delta^2 S\left(u^0;\delta u(\cdot)\right) &:= \delta x^T(t_1)\varphi_{xx}\left(x^0(t_1)\right)\delta x(t_1)-\int_{t_0}^{t_1}\left\{\delta x^T(t)H_{xx}(t)\delta x(t)\right.\\ &+\delta u^T(t)H_{uu}(t)\delta u(t)+\delta u^T(t-h)H_{\upsilon\upsilon}(t)\delta u(t-h)+2\left[\delta x^T(t)H_{xu}(t)\delta u(t)\right.\\ &\left.\left.+\delta x^T(t)H_{x\upsilon}(t)\delta u(t-h)+\delta u^T(t)H_{u\upsilon}(t)\delta u(t-h)\right]\right\}dt,\\ \delta u(\cdot)&\in \tilde{C}^{+}\left(I_1,R^r\right),\delta u(t)=0,\ t\in I_0,\end{aligned} \tag{2.3}$$

where $\delta^1 S\left(u^0;\delta u(\cdot)\right)$ and $\delta^2 S\left(u^0;\delta u(\cdot)\right)$ are, respectively, the first and the second variations of the functional $S(u)$ at the point $u^0(\cdot)$; $H(\psi,x,u,\upsilon,t):=\psi^T f(x,u,\upsilon,t)$, $H(t):=H\left(\psi^0(t),x^0(t),u^0(t),\upsilon^0(t),t\right), H_\mu(t):=H_\mu\left(\psi^0(t),x^0(t),u^0(t),\upsilon^0(t),t\right)$, $H_{\mu\nu}(t):=H_{\mu\nu}\left(\psi^0(t),x^0(t),u^0(t),\upsilon^0(t),t\right),\ t\in I$, $\mu,\nu\in\{x,u,v\}$; $\delta u(\cdot)$ is the variation of the control $u^0(\cdot)$, while $\delta x(\cdot)$ is the corresponding variation of the trajectory $x^0(t)$, $t\in I$, which $\delta x(\cdot)$ is the solution of the system

$$\begin{aligned}&\delta\dot{x}(t)=f_x(t)\delta x(t)+f_u(t)\delta u(t)+f_\upsilon(t)\delta u(t-h),\ t\in I,\\ &\delta x(t_0)=0\ ,\ \delta u(t)=0,\ t=I_0,\end{aligned} \tag{2.4}$$

where $f_\mu(t):=f_\mu\left(x^0(t),u^0(t),u^0(t-h),t\right)$, $t\in I$ and $\mu\in\{x,u,v\}$, while the vector function $\psi^0(\cdot)$ is the solution of the conjugate system

$$\dot{\psi}^0(t)=-H_x(t),\ t\in I,\quad \psi^0(t_1)=-\varphi_x\left(x^0(t_1)\right). \tag{2.5}$$

Below, we consider that the following conditions are fulfilled:

$$H(t)=0,\ H_\mu(t)=0,\ H_{\mu\nu}(t)=0,\ \text{for}\ t>t_1\ \text{and}\ \mu,\nu\in\{x,u,\upsilon\}\ . \tag{2.6}$$

http://dx.doi.org/10.5772/64225

If $\left(u^0(\cdot), x^0(\cdot)\right)$ is an optimal process, then, by definition of an admissible control and taking into consideration (2.2)–(2.4) from (2.1), proceeding the same way as in [27, p. 53], we obtain the classical necessary conditions of optimality (analogs of the Euler equation and Legendre-Clebsch condition) [10, 43], that is, the following relations are valid:

a.
$$H_u(t) + \chi(t) H_v(t+h) = 0, \quad \forall t \in I; \tag{2.7}$$

b.
$$\tilde{u}^T \left[H_{uu}(t) + \chi(t) H_{vv}(t+h) \right] \tilde{u} \le 0, \quad \forall t \in I, \forall \tilde{u} \in R^r; \tag{2.8}$$

c. $H_v(t_1) = 0, \quad \tilde{u}^T \left[H_{uu}(t_1 - h) + H_{vv}(t_1) \right] \tilde{u} \le 0,$ for all $\tilde{u} \in R^r$, if optimal control $u^0(\cdot)$ is continuous at the points $t = t_1 - ih, i = 1, 2$. Here, $\chi(\cdot)$ is the characteristic function of the set $\left[t_0, t_1 - h\right)$.

It should be noted that the optimality condition (c) is the corollary of conditions (a) and (b).

Definition 2.1. An admissible control $u^0(t), \quad t \in I$, satisfying conditions (2.7) and (2.8), is called singular (in classical sense) if

$$\operatorname{rang}\left[H_{uu}(t) + \chi(t) H_{vv}(t+h) \right] = r_1 < r, \ \forall t \in I.$$

In this case, the set I is called a singular plot for an admissible control $u^0(\cdot)$. The main goal of this chapter is to study such singular controls.

Let $u = (p, q)^T, \ v = (\tilde{p}, \tilde{q})^T$, where $p, \tilde{p} \in R^{r_0}, \ q, \tilde{q} \in R^{r_1}, \ r_0 + r_1 = r$. Without loss of generality [[27], p. 138], we assume that the singularity to the control $u^0(\cdot)$ is delivered by a vector component $p \in R^{r_0}$, that is,

$$H_{pp}(t) + \chi(t) H_{\tilde{p}\tilde{p}}(t+h) = 0, \ t \in I. \tag{2.9}$$

Note that the general inequality (2.8) implies the equality-type optimality condition for a singular (in classical sense) control $u^0(\cdot)$:

$$H_{pq}(t) + \chi(t) H_{\tilde{p}\tilde{q}}(t+h) = 0, \ t \in I. \tag{2.10}$$

Proposition 2.1. Let assumptions (A1) and (A2) be fulfilled, the admissible control $u^0(\cdot) = (p(\cdot), q(\cdot))^T$ be singular (in the classical sense) and condition (2.9) be fulfilled along it. Let also the variations $\delta u(t) = \left(\delta_0 p(t), \ \delta q(t)\right)^T \in \tilde{C}^+\left(I_1, R^r\right)$ be non-zero only on $[\theta, \theta + \varepsilon)$,

where $\theta \in [t_0, t_1)$ and $\varepsilon \in (0,\ \varepsilon_0)$, with the number $\varepsilon_0 \in (0, h)$ be such that (1) if $\theta \in [t_0, t_1 - h)$, then $\varepsilon_0 < t_1 - \theta - h$ and (2) if $\theta \in [t_1 - h, t_1)$, then $\varepsilon_0 < t_1 - \theta$. Then, (a) the variational system (2.4) becomes

$$\begin{cases} \delta\dot{x}(t) = f_x(t)\delta x(t) + f_p(t)\delta_0 p(t) + f_q(t)\delta q(t) + f_{\tilde{p}}(t)\delta_0 p(t-h) \\ \qquad\qquad + f_{\tilde{q}}(t)\delta q(t-h), \quad t \in [\theta, t_1], \\ \delta x(t) = 0,\ t \in [t_0, \theta],\ \delta_0 p(t) = 0,\ \delta q(t) = 0, \quad t \in [t_0 - h, \theta); \end{cases} \tag{2.11}$$

(b) the following representation is valid for the second variation (2.3):

$$\begin{aligned} \delta^2 S(u^0; \delta u(\cdot)) = {} & \delta x^T(t_1)\varphi_{xx}(x^0(t_1))\delta x(t_1) - \int_\theta^{t_1} \delta x^T(t) H_{xx}(t)\delta x(t)dt - \\ & -2\int_\theta^{\theta+\varepsilon} \{[\delta x^T(t)H_{xp}(t) + \delta x^T(t+h)H_{x\tilde{p}}(t+h)]\delta_0 p(t) \\ & + [\delta x^T(t)H_{xq}(t) + \delta x^T(t+h)H_{x\tilde{q}}(t+h)]\delta q(t)\}dt - \\ & -2\int_\theta^{\theta+\varepsilon} \delta_0 p^T(t)[H_{pq}(t) + H_{\tilde{p}\tilde{q}}(t+h)]\delta q(t)dt - \\ & -\int_\theta^{\theta+\varepsilon} \delta q^T(t)[H_{qq}(t) + H_{\tilde{q}\tilde{q}}(t+h)]\delta q(t)dt, \ \forall \varepsilon \in (0, \varepsilon_0). \end{aligned} \tag{2.12}$$

Proof. To prove (a), it suffices to consider the definition of the variation $\delta u(\cdot) = (\delta_0 p(\cdot), \delta q(\cdot))^T$ in (2.4). The proof of (b) follows directly from (2.3), in view of (2.6), (2.9), (2.11), and the definition of the variation $\delta u(\cdot) = (\delta_0 p(\cdot), \delta q(\cdot))^T$.

3. Transformation of the second variation of the functional by means of modified variant of matrix impulse method (when studying singular (in the sense of Definition 2.1) of controls)

Let conditions (A1) and (A2) be fulfilled and along the singular control $u^0(\cdot)$ the equality (2.9) hold. Use Proposition 2.1. Let the variation $\delta u(t) = (\delta_0 p(t), \delta q(t))^T \in \sim C^+(I_1, R^r)$ have the form:

$$\delta_0 p(t) = \begin{cases} \xi, \ t \in [\theta, \theta+\varepsilon), \ \varepsilon \in (0, \varepsilon_0), \\ 0, \ t \in I_1 \setminus [\theta, \theta+\varepsilon), \end{cases} \qquad \delta q(t) = 0, t \in I, \tag{3.1}$$

http://dx.doi.org/10.5772/64225

where $\xi \in E^{r_0}$, $\theta \in [t_0, t_1)$, and the number ε_0 was defined in Proposition 2.1.

Along the singular control $u^0(\cdot) = (p(\cdot), q(\cdot))^T$ satisfying condition (2.9), taking into account (3.1), formula (2.12) takes the form:

$$\delta^2 S(u^0; \delta u(\cdot)) = \delta x^T(t_1)\varphi_{xx}(x^0(t_1))\delta x(t_1) - \Delta_1^* - 2\Delta_2^*, \tag{3.2}$$

where

$$\Delta_1^* = \int_\theta^{t_1} \delta x^T(t) H_{xx}(t)\delta x(t)dt, \Delta_2^* = \int_\theta^{\theta+\varepsilon} \left[\delta x^T(t) H_{xp}(t) + \delta x^T(t+h) H_{x\tilde{p}}(t+h)\right]\xi dt,$$

where $\delta x(t)$, $t \in I$ is the solution of the system (2.11).

By the Cauchy formula, we have

$$\delta x(t) = \begin{cases} \int_\theta^t \lambda(s,t)\left[f_p(s)\delta_0 p(s) + f_{\tilde{p}}(s)\delta_0 p(s-h)\right]ds, \ t \in (\theta, t_1] \\ 0, \quad t \in [t_0, \theta], \end{cases} \tag{3.3}$$

where $\lambda(s,t)$, $(s,t) \in I \times I$ is the solution of the system

$$\lambda_t(s,t) = f_x(t)\lambda(s,t), \quad t_0 \le s < t \le t_1, \tag{3.4}$$

$\lambda(s,t) = 0$, $s > t$, $\lambda(s,s) = E$ (E is a unit $n \times n$ matrix).

As (A2) and $u^0(\cdot) \in \tilde{C}^+(I_1, R^r)$ are fulfilled, then by (3.1) and (3.4) and for all $\theta \in [t_0, t_1)$, from (3.3) we get

$$\delta x(t) = \begin{cases} 0, & t \in [t_0, \theta], \\ (t-\theta)\lambda(\theta,t) f_p(\theta)\xi + o(t-\theta), & t \in (\theta, \theta+\varepsilon), \\ \varepsilon\lambda(\theta,t) f_p(\theta)\xi + o(\varepsilon), & t \in [\theta+\varepsilon, \theta+h) \cap I, \\ \varepsilon\lambda(\theta,t) f_p(\theta)\xi + (t-\theta-h)\chi(\theta)\lambda(\theta+h,t) f_{\tilde{p}}(\theta+h)\xi & \\ +o(t-\theta-h), & t \in [\theta+h, \theta+h+\varepsilon) \cap I, \\ \varepsilon\left[\lambda(\theta,t) f_p(\theta) + \chi(\theta)\lambda(\theta+h,t) f_{\tilde{p}}(\theta+h)\right]\xi + o(\varepsilon), & t \in [\theta+h+\varepsilon, t_1] \cap I \end{cases} \tag{3.5}$$

where $\chi(\cdot)$ is the characteristic function of the set $[t_0, t_1 - h)$; $o(\tau)/\tau \to 0$, as $\tau \to 0$.

By (2.6) and (3.5) and taking into account $\lambda(s,s) = E$ and $\lambda(s,t) = 0$ for $s > t$, we calculate separate terms of (3.2). As a result, after simple reasoning, we get

$$\begin{aligned}\delta x^T(t_1)\varphi_{xx}\left(x^0(t)\right)\delta x(t_1) = \varepsilon^2\xi^T\Big[&f_p^T(\theta)\lambda^T(\theta,t_1)\varphi_{xx}\left(x^0(t)\right)\lambda(\theta,t_1)f_p(\theta)+\\ &+2\chi(\theta)f_p^T(\theta)\lambda^T(\theta,t_1)\varphi_{xx}\left(x^0(t_1)\right)\lambda(\theta+h,t_1)f_{\tilde{p}}(\theta+h)+\\ &+\chi(\theta)f_{\tilde{p}}^T(\theta+h)\lambda^T(\theta+h,t_1)\varphi_{xx}\left(x^0(t_1)\right)\lambda(\theta+h,t_1)f_{\tilde{p}}(\theta+h)\Big]\xi+o\left(\varepsilon^2\right),\end{aligned} \tag{3.6}$$

$$\begin{aligned}\Delta_1^* = \varepsilon^2\xi^T\int_\theta^{t_1}\Big[&f_p^T(\theta)\lambda^T(\theta,t)H_{xx}(t)\lambda(\theta,t)f_p(\theta)\\ &+2\chi(\theta)f_p^T(\theta)\lambda^T(\theta,t)H_{xx}(t)\lambda(\theta+h,t)f_{\tilde{p}}(\theta+h)\\ &+\chi(\theta)f_{\tilde{p}}^T(\theta+h)\lambda^T(\theta+h,t)H_{xx}(t)\lambda(\theta+h,t)f_{\tilde{p}}(\theta+h)\Big]dt\xi+o\left(\varepsilon^2\right),\end{aligned} \tag{3.7}$$

$$\begin{aligned}\Delta_2^* &= \xi^T\int_\theta^{\theta+\varepsilon}\Big[f_p^T(\theta)\lambda^T(\theta,t)H_{xp}(t)(t-\theta)+\chi(\theta)\left(\varepsilon f_p^T(\theta)\lambda^T(\theta,t+h)H_{x\tilde{p}}(t+h)\right.\\ &\left.+f_{\tilde{p}}^T(\theta+h)\lambda^T(\theta+h,t+h)H_{x\tilde{p}}(t+h)(t-\theta)\right)\Big]dt\xi+o\left(\varepsilon^2\right)\\ &= \frac{\varepsilon^2}{2}\xi^T\Big[f_p^T(\theta)H_{xp}(\theta)+2\chi(\theta)f_p^T(\theta)\lambda^T(\theta,\theta+h)H_{x\tilde{p}}(\theta+h)\\ &+\chi(\theta)f_{\tilde{p}}^T(\theta+h)H_{x\tilde{p}}(\theta+h)\Big]\xi+o\left(\varepsilon^2\right).\end{aligned} \tag{3.8}$$

Following [10, 14, 17], we consider the matrix functions

$$\Psi(s,\tau) = \int_{t_0}^{t_1}\lambda^T(s,t)H_{xx}(t)\lambda(\tau,t)dt - \lambda^T(s,t_1)\varphi_{xx}\left(x^0(t_1)\right)\lambda(\tau,t_1), \quad (s,\tau)\in I\times I, \tag{3.9}$$

$$M_0[p,\tilde{p}](s,\tau) = f_p^T(s)\lambda^T(s,\tau)H_{x\tilde{p}}(\tau)+f_p^T(s)\Psi(s,\tau)f_{\tilde{p}}(\tau),\ (s,\tau)\in I\times I. \tag{3.10}$$

where $\lambda(\cdot\,,\cdot)$ is the solution of the system (3.4).

Thus, substituting (3.6)–(3.8) in (3.2), allowing for (3.9), (3.10) and equality $\lambda(s,t) = 0$, for $s > t$, $(s,t)\in I\times I$, we get the validity of the following statement.

http://dx.doi.org/10.5772/64225

Proposition 3.1. Let conditions (A1) and (A2) be fulfilled, and the admissible control $u^0(\cdot) = (p(\cdot), q(\cdot))^T$ be singular (in the classic sense) and the condition (2.9) be fulfilled along it. Then, for each $\theta \in [t_0, t_1)$ and for all $\xi \in R^{r_0}$ the following expansion is valid:

$$\begin{aligned}\delta^2 S(u^0; \delta u(\cdot)) = -\varepsilon^2 \xi^T \{M_0[p,p](\theta,\theta) + 2\chi(\theta) M_0[p,\tilde{p}](\theta,\theta+h) \\ + \chi(\theta) M_0[\tilde{p},\tilde{p}](\theta+h,\theta+h)\}\xi + o(\varepsilon^2), \ \forall \varepsilon \in (0,\varepsilon_0),\end{aligned} \tag{3.11}$$

where the number ε_0 was defined above (see Proposition 2.1), $\chi(\cdot)$ is the characteristic function of the set $[t_0, t_1 - h)$ and matrix functions $M_0[p,p](\theta,\theta)$, $M_0[p,\tilde{p}](\theta,\theta+h)$, $M_0[\tilde{p},\tilde{p}](\theta+h,\theta+h)$ that are defined by (3.10).

4. Transformation of the second variation of the functional by means of modified variant of variations transformation method

4.1. Expansion of the second variation $\delta^2 S(u^0; \delta u(\cdot))$ in Kelley-type variation (first-order transformation)

Let $u^0(\cdot)$ be a singular control satisfying condition (2.9), and assumptions (A1), (A3), and (A4) be fulfilled. Now, we proceed to generalize and apply the variation transformation method [13].

Introduce the following set dependent on the admissible control $u^0(\cdot)$:

$$\begin{aligned}I^* := I(u^0(\cdot)) = \{\theta \in [t_1 - h, t_1) : \text{ the derivative } \dot{u}^0(\cdot) \text{ is continuous} \\ \text{or continuous from the right at the point } \theta \text{ and } \theta - h\} \cup \{\theta \in [t_0, t_1 - h) : \text{the derivative} \\ \dot{u}^0(\cdot) \text{ is continuous or continuous from the right at the points } \theta \text{ and } \theta \pm h\}.\end{aligned} \tag{4.1}$$

The following properties are obvious: (1) $I \backslash I^*$ is a finite set and $t_1 \overline{\in} I^*$; (2) for every $\theta \in I^*$, there exists a sufficiently small number $\hat{\varepsilon} > 0$ such that $[\theta, \theta + \hat{\varepsilon}) \cup [\theta + h, \theta + h + \hat{\varepsilon}) \cap I \subset I^*$; and (3) by (1.2), (1.3), and (2.5), the derivatives $\dot{x}^0(\cdot)$, $\dot{\psi}^0(\cdot)$ are continuous or continuous from the right at every $\theta \in I^*$. These properties are important for our further reasoning, and we call them properties of the set I^*.

Require that the variation $\delta u(\cdot) = (\delta_0 p(\cdot), \delta q(\cdot))^T$ satisfies additionally the following conditions as well:

$$\int_{\theta}^{\theta+\varepsilon} \delta_0 p(t)dt = 0, \theta \in I^*, \delta_0 p(t) = 0,\ \delta q(t) = 0,\ t \in I_1 \setminus [\theta, \theta+\varepsilon),\ \varepsilon \in (0, \varepsilon^*), \tag{4.2}$$

where $\varepsilon^* = \min\{\varepsilon_0, \hat{\varepsilon}\}$, and $\varepsilon_0, \hat{\varepsilon}$ were defined above.

Make a passage from the variation $\delta u(t) = \left(\delta_0 p(t), \delta q(t)\right)^T$, $t \in I_1$, satisfying (4.2), to a new variation $\delta_1 u(t) = \left(\delta_1 p(t), \delta q(t)\right)^T, t \in I_1$, where

$$\delta_1 p(t) = \int_{\theta}^{t} \delta_0 p(\tau)d\tau,\ t \in I_1. \tag{4.3}$$

Obvious,

$$\delta_1 p(t) = 0, t \in I_1 \setminus (\theta, \theta+\varepsilon). \tag{4.4}$$

Transform the variation of the trajectory as well: in place of $\delta x(t)$, $t \in I$, consider the function $\delta_1 x(t)$, $t \in I$:

$$\delta_1 x(t) = \delta x(t) - g_0[p](t)\delta_1 p(t) - g_0[\tilde{p}](t)\delta_1 p(t-h), t \in I, \tag{4.5}$$

where

$$g_0[\mu](t) := f_\mu(t), t \in I,\ \mu \in \{p, \tilde{p}\}. \tag{4.6}$$

As assumptions (A3) and (A4) are fulfilled, then by virtue of property of the set I^* we easily have: the function $\delta_1 x(t)$, $t \in I$ is continuous and $\delta_1 \dot{x}(t) \in \tilde{C}(I, R^n)$.

By direct differentiation, allowing for (A3), (A4) and (2.11), (4.3), (4.4) from (4.5) we obtain that $\delta_1 x(t)$, $t \in I$ is the solution of the system

$$\begin{aligned} \delta_1 \dot{x}(t) = {} & f_x(t)\delta_1 x(t) + g_1[p](t)\delta_1 p(t) + g_1[\tilde{p}](t)\delta_1 p(t-h) \\ & + f_q(t)\delta q(t) + f_{\tilde{q}}(t)\delta q(t-h), t \in [\theta, t_1], \end{aligned} \tag{4.7}$$

$$\delta_1 x(t) = 0, t \in [t_0, \theta],\ \delta_1 p(t) = 0, \delta q(t) = 0, \quad t \in [t_0 - h, \theta), \tag{4.8}$$

http://dx.doi.org/10.5772/64225

where

$$g_1[\mu](t) := f_x(t)g_0[\mu](t) - \frac{d}{dt}g_0[\mu](t),\ t \in I,\ \mu \in \{p, \tilde{p}\}. \tag{4.9}$$

Now, let us write down the second variation (2.12) in terms of new variables. By (4.4) from (4.5), we have $\delta x(t_1) = \delta_1 x(t_1)$. According to this property and (4.2)–(4.6), for any $\varepsilon \in (0, \varepsilon^*)$ the second variation (2.12), after simple reasoning takes a new form

$$\delta^2 S(u^0; \delta u(\cdot)) = \sum_{i=1}^{4} \Delta_i, \tag{4.10}$$

where

$$\begin{aligned}\Delta_1 := {} & \delta_1 x^T(t_1)\varphi_{xx}\left(x^0(t_1)\delta_1 x(t_1)\right) - \int_\theta^{t_1} \delta_1 x^T(t) H_{xx}(t)\delta_1 x(t)\,dt \\ & -2\int_\theta^{\theta+\varepsilon} \left[\delta_1 x^T(x) H_{xx}(t) g_0[p](t) + \delta_1 x^T(t+h) H_{xx}(t+h) g_0[\tilde{p}](t+h)\right]\delta_1 p(t)\,dt - \\ & -2\int_\theta^{\theta+\varepsilon} \left[\delta_1 x^T(x) H_{xq}(t) + \delta_1 x^T(t+h) H_{x\tilde{q}}(t+h)\right]\delta q(t)\,dt,\end{aligned} \tag{4.11}$$

$$\begin{aligned}\Delta_2 := {} & -\int_\theta^{\theta+\varepsilon} \{\delta_1 p^T(t)\left[g_0^T[p](t) H_{xx}(t) g_0[p](t) + g_0^T[\tilde{p}](t+h) H_{xx}(t+h) g_0[\tilde{p}](t+h)\right]\delta_1 p(t) \\ & +2\delta_1 p^T(t)\left[g_0^T[p](t) H_{xq}(t) + g_0^T[\tilde{p}](t+h) H_{x\tilde{q}}(t+h)\right]\delta q(t) \\ & +2\delta_0 p^T(t)\left[H_{pq}(t) + H_{\tilde{p}\tilde{q}}(t+h)\right]\delta q(t) + \delta q^T(t)\left[H_{qq}(t) + H_{\tilde{q}\tilde{q}}(t+h)\right]\delta q(t)\} dt,\end{aligned} \tag{4.12}$$

$$\Delta_3 := -2\int_\theta^{\theta+\varepsilon} \left[\delta_1 x^T(t) H_{xp}(t) + \delta_1 x^T(t+h) H_{x\tilde{p}}(t+h)\right]\delta_0 p(t)dt, \tag{4.13}$$

$$\Delta_4 := -2\int_\theta^{\theta+\varepsilon} \delta_1 p^T(t)\left[g_0^T[p](t) H_{xp}(t) + g_0^T[\tilde{p}](t+h) H_{x\tilde{p}}(t+h)\right]\delta_0 p(t)dt. \tag{4.14}$$

In the obtained representation, taking into account (A3), (A4), (4.2), (4.3), (4.7), (4.8), (4.13), (4.14) and the property of the set I^*, we transform Δ_3, Δ_4 by integration by parts. Then, we have

$$\Delta_3 := 2\int_{\theta}^{\theta+\varepsilon}\Big\{\delta_1 x^T(t)\Big[\frac{d}{dt}\big(H_{xp}(t)\big)+f_x^T(t)H_{xp}(t)\Big]$$
$$+\delta_1 x^T(t+h)\Big[\frac{d}{dt}\big(H_{x\tilde{p}}(t+h)\big)+f_x^T(t+h)H_{x\tilde{p}}(t+h)\Big]\Big\}\delta_1 p(t)dt$$
$$+2\int_{\theta}^{\theta+\varepsilon}\delta_1 p^T(t)\big[g_1^T[p](t)H_{xp}(t)+g_1^T[\tilde{p}](t+h)H_{x\tilde{p}}(t+h)\big]\delta_1 p(t)dt$$
$$+2\int_{\theta}^{\theta+\varepsilon}\delta_1 p^T(t)\big[H_{xp}(t)f_q(t)+H_{x\tilde{p}}^T(t+h)f_{\tilde{q}}(t+h)\big]\delta q(t)dt,$$
$$\Delta_4 := \int_{\theta}^{\theta+\varepsilon}\delta_0 p^T(t)\big[Q_0[p](t)+Q_0[\tilde{p}](t+h)\big]\delta_1 p(t)dt$$
$$+\int_{\theta}^{\theta+\varepsilon}\delta_1 p^T(t)\Big[\frac{d}{dt}\big(g_0^T[p](t)H_{xp}(t)\big)+\frac{d}{dt}\big(g_0^T[\tilde{p}](t+h)H_{x\tilde{p}}(t+h)\big)\Big]\delta_1 p(t)dt,$$

where $g_1[\mu](\cdot), \mu \in \{p, \tilde{p}\}$ is defined by (2.19),

$$Q_0[\mu](t) := g_0^T[\mu](t)H_{x\mu}(t) - H_{x\mu}^T(t)g_0[\mu](t), \quad t \in I,\ \mu \in \{p, \tilde{p}\}. \tag{4.15}$$

By substituting these relations in (4.10), after elementary transformations considering (4.11) and (4.12), we arrive at the validity of the following statement.

Proposition 4.1. Let assumptions (A1), (A3), (A4), and conditions (2.6) be fulfilled. Also, let the functions $g_0[\mu](\cdot)$, $g_1[\mu](\cdot)$, $Q_0[\mu](\cdot)$ be defined by (4.6), (4.9), and (4.15), respectively, and $\delta_1 x(t)$, $t \in I$ be the solution of the system (4.7) and (4.8). Then along the singular control $u^0(\cdot)$, satisfying condition (2.9), and on the variations $\delta u(t) = \big(\delta_0 p(t), \delta q(t)\big)^T$, $t \in I_1$ satisfying (4.2), (4.3), the following representation (first-order transformation) is valid:

$$\delta^2 S\big(u^0; \delta u(\cdot)\big) = \Delta_1^{(2)} S\big(u^0; \delta_1 p, \delta q, \delta_1 x, \varepsilon\big) + \Delta_2^{(2)} S\big(u^0; \delta_0 p, \delta_1 p, \delta q, \varepsilon\big),\ \forall \varepsilon \in \big(0, \varepsilon^*\big). \tag{4.16}$$

Here

$$\Delta_1^{(2)} S\big(u^0; \delta_1 p, \delta q, \delta_1 x, \varepsilon\big) = \delta_1 x^T(t_1)\varphi_{xx}\big(x^0(t_1)\big)\delta_1 x(t_1) - \int_{\theta}^{t_1}\delta_1 x^T(t)H_{xx}(t)\delta_1 x(t)dt$$
$$-2\int_{\theta}^{\theta+\varepsilon}\Big\{\big[\delta_1 x^T(t)G_1[p](t) + \delta_1^T x(t+h)G_1[\tilde{p}](t+h)\big]\delta_1 p(t) \tag{4.17}$$
$$+\big[\delta_1 x^T(t)H_{xq}(t) + \delta_1 x^T(t+h)H_{x\tilde{q}}(t+h)\big]\delta q(t)\Big\}dt$$

http://dx.doi.org/10.5772/64225

$$\begin{aligned}\Delta_2^{(2)}S\left(u^0;\delta_0 p,\delta_1 p,\delta q,\varepsilon\right)=\int_{\theta}^{\theta+\varepsilon}\Big\{&\delta_1 p^T(t)\left[L_1[p](t)+L_1[\tilde{p}](t+h)\right]\delta_1 p(t)\\&+2\delta_1 p^T(t)\left[P_1[p,q](t)+P_1[\tilde{p},\tilde{q}](t+h)\right]\delta q(t)+\\&\delta_0 p^T(t)\left[Q_0[p](t)+Q_0[\tilde{p}](t+h)\right]\delta_1 p(t)\\&-2\delta_0 p^T(t)\left[H_{pq}(t)+H_{\tilde{p}\tilde{q}}(t+h)\right]\delta q(t)\\&-\delta q^T(t)\left[H_{qq}(t)+H_{\tilde{q}\tilde{q}}(t+h)\right]\delta q(t)\Big\}dt,\end{aligned}\tag{4.18}$$

where ε^* was defined above (see (4.2)),

$$G_1[\mu](t):=H_{xx}(t)g_0[\mu](t)-f_x^T(x)H_{x\mu}(t)-\frac{d}{dt}H_{x\mu}(t),\ t\in I,\mu\in\{p,\tilde{p}\},\tag{4.19}$$

$$\begin{aligned}P_1[p,q](t)&:=H_{xp}^T(t)f_q(t)-g_0^T[p](t)H_{xq}(t),t\in I,\\P_1[\tilde{p},\tilde{q}](t)&:=H_{x\tilde{p}}^T(t)f_{\tilde{q}}(t)-g_0^T[\tilde{p}](t)H_{x\tilde{q}}(t),t\in I,\end{aligned}\tag{4.20}$$

$$\begin{aligned}&L_1[\mu](t):=-g_0^T[\mu](t)H_{xx}(t)g_0[\mu](t)+2g_1^T[\mu](t)H_{x\mu}(t)\\&+\frac{d}{dt}\left(g_0^T[\mu](t)H_{x\mu}(t)\right),t\in I,\mu\in\{p,\tilde{p}\}.\end{aligned}\tag{4.21}$$

4.2. Higher-order transformation

Let $\left(u^0(\cdot),x^0(\cdot)\right)$ be some process, where $u^0(\cdot)$ is a singular control satisfying condition (2.9), and assumptions (A1), (A5), and (A6) be fulfilled. Introduce the matrix functions calculated along the process $\left(u^0(\cdot),x^0(\cdot)\right)$ and determined by the following recurrent formulas:

$$\begin{aligned}&g_{i+1}[\mu](t)=f_x(t)g_i[\mu](t)-\frac{d}{dt}g_i[\mu](t),\\&g_0[\mu](t):=f_\mu(t),\ t\in I,\ \mu\in\{p,\tilde{p}\},i=0,1,...,\end{aligned}\tag{4.22}$$

$$\begin{aligned}&G_{1+1}[\mu](t)=H_{xx}(t)g_i[\mu](t)-f_x^T(t)G_i[\mu](t)-\frac{d}{dt}G_i[\mu](t),\\&G_0[\mu](t):=H_{x\mu}(t),\ \ t\in I,\ \mu\in\{p,\tilde{p}\},\ i=0,1,...\end{aligned}\tag{4.23}$$

Furthermore, similar to (4.15), (4.20), (4.21), and (3.10), consider the functions

$$\begin{aligned} &P_{i+1}[p,q](t) = G_i^T[p](t) f_q(t) - g_i^T[p](t) H_{xq}(t), P_0[p,q](t) := H_{pq}(t), t \in I, i = 0,1,..., \\ &P_{i+1}[\tilde{p},\tilde{q}](t) = G_i^T[\tilde{p}](t) f_{\tilde{q}}(t) - g_i^T[\tilde{p}](t) H_{x\tilde{q}}(t), P_0[\tilde{p},\tilde{q}](t) := H_{\tilde{p}\tilde{q}}(t), t \in I, \end{aligned} \tag{4.24}$$

$$Q_i[\mu](t) = g_i^T[\mu](t) G_i[\mu](t) - G_i^T[\mu](t) g_i[\mu](t), \ \mu \in \{p,\tilde{p}\}, t \in I, i = 0,1,..., \tag{4.25}$$

$$\begin{aligned} &L_{i+1}[\mu](t) = -g_i^T[\mu](t) H_{xx}(t) g_i[\mu](t) + 2g_{i+1}^T[\mu](t) G_i[\mu](t) + \frac{d}{dt}\left(g_i^T[\mu](t) G_i[\mu](t)\right), \\ &L_0[\mu](t) := H_{\mu\mu}(t), \ \mu \in \{p,\tilde{p}\}, t \in I, i = 0,1,..., \end{aligned} \tag{4.26}$$

$$\begin{aligned} &M_i[p,\tilde{p}](s,\tau) := g_i^T[p](s) \lambda^T(s,\tau) G_i[\tilde{p}](\tau) \\ &+ g_i^T[p](s) \Psi(s,\tau) g_i[\tilde{p}](\tau), (s,\tau) \in I \times I, \quad i = 0,1,..., \end{aligned} \tag{4.27}$$

where $\lambda(\cdot)$ and $\Psi(\cdot)$ are determined by (3.4) and (3.9), respectively.

Similar to I^*, we introduce the set I^{**} when assumption (A6) is fulfilled:

$I^{**} := I\left(u^0(\cdot)\right) = \{\theta \in [t_1 - h, t_1)$: the admissible control $u^0(\cdot)$ is sufficiently smooth or sufficiently smooth from the right at the points θ and $\theta - h\} \cup \{\theta \in [t_0, t_1 - h)$: the admissible control $u^0(\cdot)$ is sufficiently smooth or sufficiently smooth

$$\text{from the right at the points } \theta \text{ and } \theta \pm h\}. \tag{4.28}$$

The following obvious properties hold: (1) $I \backslash I^{**}$ is a finite set, and $t_1 \overline{\in} I^{**}$, also $I^{**} \subset I^*$; (2) for every $\theta \in I^{**}$ there exists a sufficiently small number $\tilde{\varepsilon} > 0$, such that $[\theta, \theta + \hat{\varepsilon}) \cup [\theta + h, \theta + h + \hat{\varepsilon}) \cap I \subset I^*$, furthermore, (3) by (A5), (A6), (1.2), (1.3), and (2.5), the functions $x^0(\cdot), \psi^0(\cdot)$ are continuous and sufficiently smooth or sufficiently smooth from the right at every point $\theta \in I^{**}$. These properties are important at the next reasoning and we call them the properties of the set I^{**}.

Let us consider a variation $\delta u(t) = \left(\delta_0 p(t), \delta q(t)\right)^T, \ t \in I_1$ that in addition satisfies the following conditions as well:

$$\begin{aligned} &\delta_0 p(t) = 0, \delta q(t) = 0, t \in I_1 \setminus [\theta, \theta + \varepsilon), \\ &\delta_1 p(t) = ... = \delta_k p(t) = 0, t \in I_1 \setminus (\theta, \theta + \varepsilon), \end{aligned} \tag{4.29}$$

where

http://dx.doi.org/10.5772/64225

$$\delta_i p(t) = \int_\theta^t \delta_{i-1} p(\tau)\, d\tau,\ t \in I_1,\ i = 1,2,...,k,\ k \in \{1,2,...\}, \tag{4.30}$$

$\theta \in I^{**}$, $\varepsilon \in (0, \varepsilon^{**})$, $\varepsilon^{**} = \min\{\varepsilon_0, \hat{\varepsilon}, \tilde{\varepsilon}\}$ ($\varepsilon_0, \hat{\varepsilon}, \tilde{\varepsilon}$ were defined above).

According to (4.30), we have

$$\delta_i p(t) = \int_\theta^t \frac{(t-\tau)^{i-1}}{(i-1)!} \delta_0 p(\tau)\, d\tau, \theta \in I^{**},\ t \in I_1,\ i = 1,2,...,k,\ k \in \{1,2,...\}. \tag{4.31}$$

The following statement is valid.

Proposition 4.2. Let assumptions (A1), (A5), (A6), and condition (2.6) be fulfilled. Furthermore, let the functions $g_i[\mu](\cdot)$, $G_i[\mu](\cdot)$, $P_i[p,q](\cdot)$, $P_i[\tilde{p},\tilde{q}](\cdot)$, $Q_i[\mu](\cdot)$ and $L_i[\mu](\cdot)$, where $\mu \in \{p, \tilde{p}\}$, $i = 0, 1, ...$, be defined by (4.22)–(4.26), and the set I^{**} be defined by (4.28). Then along the singular control $u^0(\cdot)$, satisfying condition (2.9), and on the variations $\delta u(t) = \left(\delta_0 p(t), \delta q(t)\right)^T$, $t \in I_1$ satisfying (4.29) and (4.30), the following representation (k-th order transformation, where $k \in \{1, 2, ...\}$) is valid:

$$\delta^2 S\left(u^0; \delta u\right) = \Delta_1^2 S\left(u^0; \delta_k p, \delta q, \delta_k x, \varepsilon\right) + \Delta_2^2 S\left(u^0; \delta_0 p, ..., \delta_k p, \delta q, \varepsilon\right). \tag{4.32}$$

Here

$$\begin{aligned}\Delta_1^2 S(\cdot) = {} & \delta_k x^T(t_1)\varphi_{xx}\left(x^0(t_1)\right)\delta_k x(t_1) - \int_\theta^{t_1} \delta_k x^T(t) H_{xx}(t)\delta_k x(t)\, dt \\ & -2\int_\theta^{\theta+\varepsilon} \Big\{\left[\delta_k x^T(t) G_k[p](t) + \delta_k x^T(t+h) G_k[\tilde{p}](t+h)\right]\delta_k p(t) \\ & +\left[\delta_k x^T(t) H_{xq}(t) + \delta_k x^T(t+h) H_{x\tilde{q}}(t+h)\right]\delta q(t)\Big\}\, dt,\end{aligned} \tag{4.33}$$

$$\begin{aligned}\Delta_2^2 S(\cdot) = {} & \int_\theta^{\theta+\varepsilon} \Big\{\sum_{i=0}^{k-1}\Big[\delta_{i+1} p^T(t)\left(L_{i+1}[p](t) + L_{i+1}[\tilde{p}](t+h)\right)\delta_{i+1} p(t)\, dt \\ & +2\delta_{i+1} p(t)\left(P_{i+1}[p,q](t) + P_{i+1}[\tilde{p},\tilde{q}](t+h)\right)\delta q(t) \\ & +\delta_i p^T(t)\left(Q_i[p](t) + Q_i[\tilde{p}](t+h)\right)\delta_{i+1} p(t)\Big] \\ & -2\delta_0 p^T(t)\left(P_0[p,q](t) + P_0[\tilde{p},\tilde{q}](t+h)\right)\delta q(t) \\ & -\delta q^T(t)\left(H_{qq}(t) + H_{\tilde{p},\tilde{q}}(t+h)\right)\delta q(t)\Big\}\, dt,\end{aligned} \tag{4.34}$$

where $\theta \in I^{**}$, $\varepsilon \in (0, \varepsilon^{**})$ (the number ε^{**} was defined above), $\delta_k x(t)$, $t \in I$ is the solution of the system

$$\begin{aligned}&\delta_k\dot{x}(t)=f_x(t)\delta_k x(t)+g_k[p](t)\delta_k p(t)+g_k[\tilde{p}](t)\delta_k p(t-h)\\&+f_q(t)\delta q(t)+f_{\tilde{q}}(t)\delta q(t-h),t\in[\theta,t_1]\\&\delta_k x(t)=0,\ t\in[t_0,\theta],\ \delta_k p(t)=0,\ \delta q(t)=0,\ t\in[t_0-h,\theta),\ k\in\{1,2,...\},\end{aligned}\tag{4.35}$$

Proof. We carry out the proof of Proposition 4.2 by induction. For $k = 1$, Proposition 4.2 was completely proved at item 4 (see Proposition 4.1). Assume that Proposition 4.2 is valid for all the cases to $(k-1)$ inclusively, $(k \geq 2)$. We prove the validity of representation (4.32) for the case k. Let the variation $\delta u(t) = \left(\delta_0 p(t), \delta q(t)\right)^T$, $t \in I_1$ satisfies the conditions (4.29) and (4.30). Then by assumption the following representation is valid:

$$\delta^2 S(u^0;\delta u)=\Delta_1^2 S\left(u^0;\delta_{k-1}p,\delta q,\delta_{k-1}x,\varepsilon\right)+\Delta_2^2 S\left(u^0;\delta_0 p,...,\delta_{k-1}p,\delta q,\varepsilon\right).\tag{4.36}$$

Here

$$\begin{aligned}\Delta_1^2 S\left(u^0;\delta_{k-1}p,\delta q,\delta_{k-1}x,\varepsilon\right)&=\delta_{k-1}x^T(t_1)\varphi_{xx}\left(x^0(t_1)\right)\delta_{k-1}x(t_1)-\int_\theta^{t_1}\delta_{k-1}x^T(t)H_{xx}(t)\delta_{k-1}x(t)dt\\&-2\int_\theta^{\theta+\varepsilon}\Big\{\Big[\delta_{k-1}x^T(t)G_{k-1}[p](t)+\delta_{k-1}x^T(t+h)G_{k-1}[\tilde{p}](t+h)\Big]\delta_{k-1}p(t)\\&+\Big[\delta_{k-1}x^T(t)H_{xq}(t)+\delta_{k-1}x^T(t+h)H_{x\tilde{q}}(t+h)\Big]\delta q(t)\Big\}dt,\end{aligned}\tag{4.37}$$

$$\begin{aligned}&\Delta_2^2 S\left(u^0;\delta_0 p,...,\delta_{k-1}p,\delta q,\varepsilon\right)=\\&\int_\theta^{\theta+\varepsilon}\Big\{\sum_{i=0}^{k-2}\Big[\delta_{i+1}p^T(t)\Big(L_{i+1}[p](t)+L_{i+1}[\tilde{p}](t+h)\Big)\delta_{i+1}p(t)\\&+2\delta_{i+1}p^T(t)\Big(P_{i+1}[p,q](t)+P_{i+1}[\tilde{p},\tilde{q}](t+h)\Big)\delta q(t)\\&+\delta_i p^T(t)\Big(Q_i[p](t)+Q_i[\tilde{p}](t+h)\Big)\delta_{i+1}p(t)\Big]-\\&2\delta_0 p^T(t)\Big(P_0[p,q](t)+P_0[\tilde{p},\tilde{q}](t+h)\Big)\delta q(t)\\&-\delta q^T(t)\Big(H_{qq}(t)+H_{\tilde{q}\tilde{q}}(t+h)\Big)\delta q(t)\Big\}dt,\end{aligned}\tag{4.38}$$

where $G_i[\mu](\cdot)$, $P_i[p,q](\cdot)$, $P_i[\tilde{p},\tilde{q}](\cdot)$, $Q_i[\mu](\cdot)$, $L_i[\mu](\cdot)$, $\mu \in \{p,\tilde{p}\}$, $i = 0,1,...$ are defined by (4.23)-(4.26), and $\delta_{k-1}x(t), t \in I$ is the solution of the system:

http://dx.doi.org/10.5772/64225

$$\begin{aligned}&\delta_{k-1}\dot{x}(t)=f_x(t)\delta_{k-1}x(t)+g_{k-1}[p](t)\delta_{k-1}p(t)+g_{k-1}[\tilde{p}](t)\delta_{k-1}p(t-h)\\&+f_q(t)\delta q(t)+f_{\tilde{q}}(t)\delta q(t-h),\\&\delta_{k-1}x(t)=0,\ t\in[t_0,\theta],\ \delta_{k-1}p(t)=0,\ \delta q(t)=0,\ t\in[t_0-h,\theta),\ k\geq 2.\end{aligned}\tag{4.39}$$

Apply the modified variant of variations transformations method [13] to the system for $\delta_{k-1}x(t), t\in I$ and representation (4.36). According to the technique of the previous item (see item 4.1), we introduce a new variation in the following way:

$$\delta_k x(t)=\delta_{k-1}x(t)-g_{k-1}[p](t)\delta_k p(t)-g_{k-1}[\tilde{p}](t)\delta_k p(t-h),t\in I.\tag{4.40}$$

According to (4.22), (4.30), (4.31), and (4.39) from (4.40) by direct differentiation, we get the system (4.35) for $\delta_k x(t), t\in I$. Furthermore, as $\theta\in I^{**}$, then by (4.40) we get $\delta_k x(t_1)=\delta_{k-1}x(t_1)$. Taking into account this equality and by (4.29), (4.30), and (4.40) in (4.37), let us transform the representation (4.36) into new variables $\delta_k p(\cdot)$, $\delta q(\cdot)$, $\delta_k x(\cdot)$. Then,

$$\begin{aligned}\delta^2 S(u^0;\delta u)=&\delta_k x^T(t_1)\varphi_{xx}(x^0(t_1))\delta_k x(t_1)\\&-\Delta_{11}-\Delta_{12}-\Delta_{13}+\Delta_2^2 S(u^0;\delta_0 p,\ldots,\delta_{k-1}p,\delta q,\varepsilon),\end{aligned}\tag{4.41}$$

where $\Delta_2^2 S(\cdot)$ is determined by formula (4.38) as well as Δ_{i1}, $i=1,2,3$ by (4.22), (4.29), (4.30), (4.35), (2.6) are calculated in the following way:

$$\begin{aligned}\Delta_{11}=&\int_\theta^{t_1}\delta_k x^T(t)H_{xx}(t)\delta_k x(t)dt\\&+2\int_\theta^{\theta+\varepsilon}\left[\delta_k x^T(t)H_{xx}(t)g_{k-1}[p](t)+\delta_k x^T(t+h)H_{xx}(t+h)g_{k-1}[\tilde{p}](t+h)\right]\delta_k p(t)\\&+\int_\theta^{\theta+\varepsilon}\delta_k p^T(t)\Big[g_{k-1}^T[p](t)H_{xx}(t)g_{k-1}[p](t)\\&+g_{k-1}^T[\tilde{p}](t+h)H_{xx}(t+h)g_{k-1}[\tilde{p}](t+h)\Big]\delta_k p(t)dt,\end{aligned}\tag{4.42}$$

$$\begin{aligned}\Delta_{12}=&2\int_\theta^{\theta+\varepsilon}\Big\{\Big[\delta_k x^T(t)+\delta_k p^T(t)g_{k-1}^T[p](t)\Big]G_{k-1}[p](t)\\&+\Big[\delta_k x^T(t+h)+\delta_k p^T(t)g_{k-1}^T[\tilde{p}](t+h)\Big]G_{k-1}[\tilde{p}](t+h)\Big\}\delta_{k-1}p(t)dt=:\Delta_{12}^*+\Delta_{12}^{**},\end{aligned}\tag{4.43}$$

where

$$
\begin{aligned}
\Delta_{12}^{*} &:= 2\int_{\theta}^{\theta+\varepsilon}\Big[\delta_k x^T(t)G_{k-1}[p](t)+\delta_k x(t+h)G_{k-1}[\tilde{p}](t+h)\Big]\delta_{k-1}p(t)\,dt,\\
\Delta_{12}^{**} &:= 2\int_{\theta}^{\theta+\varepsilon}\delta_k p^T(t)\Big[g_{k-1}^T[p]G_{k-1}[p](t)+g_{k-1}^T[\tilde{p}](t+h)G_{k-1}[\tilde{p}](t+h)\Big]\delta_{k-1}p(t)\,dt,\\
\Delta_{13} &= 2\int_{\theta}^{\theta+\varepsilon}\Big\{\Big[\delta_k x^T(t)+\delta_k p^T(t)g_{k-1}^T[p](t)\Big]H_{xq}(t)\\
&+\Big[\delta_k x^T(t+h)+\delta_k p^T(t)g_{k-1}^T[\tilde{p}](t+h)\Big]H_{x\tilde{q}}(t+h)\Big\}\delta q(t)\,dt\\
&= 2\int_{\theta}^{\theta+\varepsilon}\Big[\delta_k x^T(t)H_{xq}(t)+\delta_k x^T(t+h)H_{x\tilde{q}}(t+h)\Big]\delta q(t)\,dt\\
&+2\int_{\theta}^{\theta+\varepsilon}\delta_k p^T(t)\Big[g_{k-1}^T[p](t)H_{xq}(t)+g_{k-1}^T[\tilde{p}](t+h)H_{x\tilde{q}}(t+h)\Big]\delta q(t)\,dt.
\end{aligned}
\tag{4.44}
$$

Taking into account (A5), (A6), (4.29), (4.30), (4.35), and the properties of the set I^{**}, let us calculate $\Delta_{12}^{*}, \Delta_{12}^{**}$. Then, applying the method of integration by parts, we have

$$
\begin{aligned}
\Delta_{12}^{*} &= -2\int_{\theta}^{\theta+\varepsilon}\Bigg\{\Bigg[\delta_k x^T(t)\Bigg(f_x^T(t)G_{k-1}[p](t)+\frac{d}{dt}\Big(G_{k-1}[p](t)\Big)\Bigg)\\
&\quad+\delta_k x^T(t+h)\Bigg(f_x^T(t+h)G_{k-1}[\tilde{p}](t+h)+\frac{d}{dt}\Big(G_{k-1}[\tilde{p}](t+h)\Big)\Bigg)\Bigg]\delta_k p(t)\\
&\quad+\delta_k p^T(t)\Big[g_k^T[p](t)G_{k-1}[p](t)+g_k^T[\tilde{p}](t+h)G_{k-1}[\tilde{p}](t+h)\Big]\delta_k p(t)\\
&\quad+\delta_k p^T(t)\Big[G_{k-1}^T[p](t)f_q(t)+G_{k-1}^T[\tilde{p}](t+h)f_{\tilde{q}}(t+h)\Big]\delta q(t)\Big\}\,dt;\\
\Delta_{12}^{**} &= -\int_{\theta}^{\theta+\varepsilon}\delta_{k-1}p^T(t)\Big[Q_{k-1}[p](t)+Q_{k-1}[\tilde{p}](t+h)\Big]\delta_k p(t)\,dt\\
&\quad-\int_{\theta}^{\theta+\varepsilon}\delta_k p^T(t)\Bigg[\frac{d}{dt}\Big(g_{k-1}^T[p](t)G_{k-1}[p](t)\Big)+\frac{d}{dt}\Big(g_{k-1}^T[\tilde{p}](t+h)G_{k-1}[\tilde{p}](t+h)\Big)\Bigg]\delta_k p(t)\,dt.
\end{aligned}
$$

At first, we substitute the last expression Δ_{12}^{*}, Δ_{12}^{**} in (4.43), and then (4.42)–(4.44) in (4.41). Then by (4.23)–(4.26), (4.33), (4.34), and (4.38), it is easy to get representation (4.32). Consequently, we get the proof for k. This completes the proof of Proposition 4.2.

5. Optimality conditions

Based on Propositions 3.1, 4.1, and 4.2, we prove the following theorem.

Theorem 5.1. Let conditions (A1), (A5), and (A6) be fulfilled, and the matrix functions $P_i[p,q](\cdot), P_i[\tilde{p},\tilde{q}](\cdot)$, $Q_i[\mu](\cdot)$, $L_i[\mu](\cdot), M_i[p,\tilde{p}](\cdot)$, $\mu \in \{p,\tilde{p}\}$, $i = 0,1,\ldots$ be defined as in

http://dx.doi.org/10.5772/64225

(4.24)–(4.27). Let also the set I^{**} be defined as in (4.28) and along the singular (in the classical sense) control $u^0(\cdot)$ the following equalities be fulfilled:

$$L_i[p](t) + \chi(t) L_i[\tilde{p}](t+h) = 0, \ \forall t \in I^{**}, i = 0,1,...,k, \ k \in \{0,1,...\}, \tag{5.1}$$

where $\chi(\cdot)$ is the characteristic function of the set $[t_0, t_1 - h)$.

Then for the optimality of the admissible control $u^0(\cdot)$, it is necessary that the relations

$$P_i[p,q](\theta) + \chi(t) P_i[\tilde{p},\tilde{q}](\theta+h) = 0, \ i = 0,1,...,k, \tag{5.2}$$

$$\begin{aligned} &\xi^T \{M_i[p,p](\theta,\theta) + 2\chi(t) M_i[p,\tilde{p}](\theta,\theta+h) \\ &+\chi(\theta) M_i[\tilde{p},\tilde{p}](\theta+h,\theta+h)\}\xi \le 0, \ i = 0,1,...,k, \end{aligned} \tag{5.3}$$

$$Q_i[p](\theta) + \chi(\theta) Q_i[\tilde{p}](\theta+h) = 0, \ i = 0,1,...,k, \tag{5.4}$$

$$\begin{aligned} &\mathcal{L}_{k+1}(\theta,\xi,\eta) := \xi^T (L_{k+1}[p](\theta) + \chi(\theta) L_{k+1}[\tilde{p}](\theta+h))\xi + \\ &+ 2\xi^T (P_{k+1}[p,q](\theta) + \chi(\theta) P_{k+1}[\tilde{p},\tilde{q}](\theta+h))\eta - \\ &- \eta^T (H_{qq}(\theta) + \chi(\theta) H_{\tilde{q}\tilde{q}}(\theta+h))\eta \ge 0, \end{aligned} \tag{5.5}$$

be fulfilled for all $\theta \in I^{**}$, $\xi \in R^{r_0}$ and $\eta \in R^{r_1}$.

Proof. Let $u^0(\cdot)$ be an optimal control. We will prove the theorem by induction. Let $k = 0$, that is, $i = 0$. Then, according to (4.24) and (2.10) we get the proof of optimality condition (5.2) for $k = 0$. The proof of optimality condition (5.3) for $k = 0$ directly follows from (3.11) allowing for (2.1) (see Proposition 3.1). Now, based on Proposition 4.1 prove the optimality conditions (5.4) and (5.5) for $k = 0$.

We first prove the validity of (5.4) for k=0.

Suppose that

$$\delta_0 p_m(t) = 0, \ \forall t \in I_1, \forall m \in \{1,2,...,r_0\} \setminus \{i,j\} \tag{5.6}$$

$$\delta_0 p_i(t) = \begin{cases} \alpha l_1\left(\dfrac{2(t-\theta)}{\varepsilon} - 1\right), & t \in [\theta, \theta+\varepsilon),\ \varepsilon \in (0, \varepsilon^{**}), \\ 0, & t \in I_1 \setminus [\theta, \theta+\varepsilon), \end{cases}$$

$$\delta_0 p_j(t) = \begin{cases} \beta l_2\left(\dfrac{2(t-\theta)}{\varepsilon} - 1\right), & t \in [\theta, \theta+\varepsilon),\ \varepsilon \in (0, \varepsilon^{**}), \\ 0, & t \in I_1 \setminus [\theta, \theta+\varepsilon), \end{cases}$$

$$\delta q(t) = 0,\ t \in I_1$$

where i, j $(i \neq j)$ are arbitrary fixed points of the set $\{1, 2, ..., r_0\}$ and $\delta_0 p_k(\cdot)$ is the k-th coordinate of the vector $\delta_0 p(\cdot)$; $\alpha, \beta \in R$ and $\theta \in I^{**}$ are arbitrary fixed points, the functions $l_1(s) = s,\ l_2(s) = \frac{3}{2}s^2 - \frac{1}{2},\ s \in [-1, 1]$ are the Legendre polynomials.

It is clear that the variation $\delta u(t) = (\delta_0 p(t), \delta q(t))^T$, $t \in I_0 \cup I =: I_1$, defined by (5.6) satisfies the condition (4.2) and, according to (5.6) the function $\delta_1 p(t),\ t \in I_1$, defined by (4.3) is of order ε, and the solution $\delta_1 x(t)$, $t \in I$ of the system (4.7), (4.8) is of order ε^2. Also, according to (4.15) it is easy to see that for every $t \in I$ the matrix $Q_0[p](t) + \chi(t) Q_0[\tilde{p}](t+h)$ is skew-symmetric. Therefore, by Proposition 4.1 and condition (2.6), considering (2.1), (4.3), (4.17), (4.18), and the properties of the set I^{**}, along the singular optimal control $u^0(\cdot)$, we have

$$\begin{aligned} \delta^2 S(u^0; \delta u(\cdot)) &= \int_\theta^{\theta+\varepsilon} \delta_0 p^T(t) \left[Q_0[p](t) + \chi(t) Q_0[\tilde{p}](t+h) \right] \delta_1 p(t)\, dt + o(\varepsilon^2) \\ &= \int_\theta^{\theta+\varepsilon} \left[q_{ij}^{(0)}(t) \delta_0 p_i(t) \delta_1 p_j(t) + q_{ji}^{(0)}(t) \delta_0 p_j(t) \delta_1 p_i(t) \right] dt + o(\varepsilon^2) \\ &= \frac{\varepsilon^2}{4} \alpha\beta \left[q_{ij}^{(0)}(\theta) - q_{ji}^{(0)}(\theta) \right] \int_{-1}^{1} l_1(s) \int_{-1}^{s} l_2(\tau)\, d\tau\, ds + o(\varepsilon^2) \\ &= -\frac{\varepsilon^2 \alpha\beta}{30} \left[q_{ij}^{(0)}(\theta) - q_{ji}^{(0)}(\theta) \right] + o(\varepsilon^2) \geq 0,\ \forall \varepsilon \in (0, \varepsilon^{**}) \end{aligned}$$

where $q_{ij}^{(0)}(\theta)$, $q_{ji}^{(0)}(\theta)$ are the elements of the matrix $Q_0[p](\theta) + \chi(\theta) Q_0[\tilde{p}](\theta + h)$.

Then, we conclude from the arbitrariness of $\alpha, \beta \in R$, $\theta \in I^{**}$ and $i, j \in \{1, 2, ..., r_0\}$, $i \neq j$ that the skew-symmetric matrix $Q_0[p](\theta) + \chi(\theta) Q_0[\tilde{p}](\theta + h)$ is also symmetric. Consequently, for

http://dx.doi.org/10.5772/64225

every $t \in I^{**}$ we have $Q_0[p](t) + \chi(t)Q_0[\tilde{p}](t+h) = 0$. This completes the proof of the optimality condition (5.4) for $k = 0$.

To prove statement (5.5) for $k = 0$, under the conditions (4.2) and (4.3), we write down the vector components of the variation $\delta u(\cdot) = \left(\delta_0 p(\cdot), \delta q(\cdot)\right)^T$ in the following form:

$$\delta_0 p(t) = \begin{cases} \xi l_1\left(\dfrac{2(t-\theta)}{\varepsilon} - 1\right), & t \in [\theta, \theta+\varepsilon), \\ 0 & t \in I_1 \setminus [\theta, \theta+\varepsilon),\ \varepsilon \in (0, \varepsilon^{**}), \end{cases}$$
$$\delta q(t) = \begin{cases} \eta \int_\theta^t l_1\left(\dfrac{2(s-\theta)}{\varepsilon} - 1\right) ds, & t \in [\theta, \theta+\varepsilon), \\ 0, & t \in I_1 \setminus [\theta, \theta+\varepsilon),\ \varepsilon \in (0, \varepsilon^{**}), \end{cases} \tag{5.7}$$

where $l_1(\tau) = \tau$, $\tau \in [-1, 1]$ is a Legendre polynomial, $\xi \in R^{r_0}$, $\eta \in R^{r_1}$, $\theta \in I^{**}$ are arbitrary fixed points.

According to (4.2), (4.3), (4.7), (4.8), and (5.7), it is easy to prove that

$$\delta_1 p(t) \sim \varepsilon, \delta q(t) \sim \varepsilon,\ t \in I_1,\ \delta_1 x(t) \sim \varepsilon^2,\ t \in I.$$

In view of the last relations and above proved condition (5.4) (for the case $k = 0$) taking into account the properties of the set I^{**} and the relations (2.1), (4.3), (4.17), (4.18), and (5.7) from (4.16), we obtain the following relation along the singular optimal control $u^0(t), t \in I_1$:

$$\begin{aligned} \delta^2 S\left(u^0; \delta u(\cdot)\right) &= \int_\theta^{\theta+\varepsilon} \left\{ \delta_1 p^T(t)\left[L_1[p](t) + \chi(t) L_1[\tilde{p}](t+h)\right]\delta_1 p(t) \right. \\ &+ 2\delta_1 p^T(t)\left[P_1[p,q](t) + \chi(t) P_1[\tilde{p},\tilde{q}](t+h)\right]\delta q(t) \\ &\left. - \delta q^T(t)\left[H_{qq}(t) + \chi(t) H_{\tilde{q}\tilde{q}}(t+h)\right]\delta q(t) \right\} dt + o\left(\varepsilon^3\right) \\ &= \frac{\varepsilon^3}{8}\left\{ \xi^T\left[L_1[p](\theta) + \chi(\theta) L_1[\tilde{p}](\theta+h)\right]\xi + 2\xi^T\left[P_1[p,q](\theta) + \chi(\theta) P_1[\tilde{p},\tilde{q}](\theta+h)\right]\eta \right. \\ &\left. - \eta^T\left[H_{qq}(\theta) + \chi(\theta) H_{\tilde{q}\tilde{q}}(\theta+h)\right]\eta \right\} \int_{-1}^{1} \left(\int_{-1}^{t} l_1(\tau) d\tau \right)^2 dt + o\left(\varepsilon^3\right) \geq 0, \forall\, \varepsilon \in (0, \varepsilon^*). \end{aligned}$$

Hence, taking into account the arbitrariness of $\theta \in I^*$, $\xi \in R^{r_0}$ and $\eta \in R^{r_1}$, we easily get the validity of the optimality condition (5.5) for $k = 0$.

Now suppose that all the statements of Theorem 5.1 are valid for $i = 1, 2, ..., k-1$ $(k \geq 2)$ as well. Prove statements (5.2)–(5.5), for $i = k$. By assumption, the inequality $\mathcal{L}_k(\theta, \xi, \eta) \geq 0$ (see (5.5) for the case *k-1*) is valid for all $\theta \in I^{**}$, $\xi \in R^{r_0}$ and $\eta \in R^{r_1}$. Hence, taking into account (5.1), we have

$$2\xi\left(P_k[p,q](\theta) + \chi(\theta)P_k[\tilde{p},\tilde{q}](\theta+h)\right)\eta$$
$$-\eta^T\left(H_{qq}(\theta) + \chi(\theta)H_{\tilde{q}\tilde{q}}(\theta+h)\right)\eta \geq 0, \forall\theta \in I^{**}, \forall\xi \in R^{r_0}, \forall\eta \in R^{r_1}.$$

From this inequality, we easily get that $P_k[p,q](\theta) + \chi(\theta)P_k[\tilde{p},\tilde{q}](\theta+h) = 0$, that is, we get the validity of optimality condition (5.2) for $i = k$.

Now, prove the validity of condition (5.3) for $i = k$. In formula (4.32), we put

$$\delta_0 p(t) = \begin{cases} \xi l_k\left(\dfrac{2(t-\theta)}{\varepsilon} - 1\right), & t \in [\theta, \theta+\varepsilon), \\ 0, & t \in I_1 \setminus [\theta+\varepsilon), \end{cases} \qquad \delta q(t) = 0,\ t \in I_1, \tag{5.8}$$

where $l_k(\tau)$, $\tau \in [-1,1]$ is the k-th Legendre polynomial, $\varepsilon \in (0, \varepsilon^{**})$ which the number ε^{**} is defined above (see (4.30)) and $\theta \in I^{**}$, $\xi \in R^{r_0}$.

Obviously, conditions (4.29) and (4.30) are fulfilled for variation (5.8).

As the conditions $L_i[p](t) + \chi(t)L_i[\tilde{p}](t) = 0$, $t \in I^{**}$, $i = \overline{0,k}$ and $Q_i[p](t) + \chi Q_i[\tilde{p}](t+h) = 0$, $t \in I^{**}$, $i = \overline{0,k-1}$, are fulfilled, then by (4.33), (4.34), and (5.8), formula (4.32) takes the form:

$$\delta^2 S(u^0; \delta u) = \delta_k x^T(t_1)\varphi_{xx}(x^0(t_1))\delta_k x(t_1) - \Delta_{1k}^{*} - 2\Delta_{1k}^{**}, \tag{5.9}$$

where

$$\Delta_{1k}^{*} = \int_{\theta}^{t_1} \delta_k x^T(t) H_{xx}(t)\delta_k x(t)\,dt, \tag{5.10}$$

$$\Delta_{1k}^{**} = \int_{\theta}^{\theta+\varepsilon} \left[\delta_k x^T(t) G_k[p](t) + \delta_k x^T(t+h) G_k[\tilde{p}](t+h)\right]\delta_k p(t)\,dt. \tag{5.11}$$

http://dx.doi.org/10.5772/64225

Here, by (4.31), (4.35), (5.8), and the Cauchy formula, $\delta_k p(\cdot)$ and $\delta_k x(\cdot)$ are determined as follows:

$$\delta_k p(t)=\begin{cases}\xi\int_\theta^t \frac{(t-s)^{k-1}}{(k-1)!} l_k\left(\frac{2(s-\theta)}{\varepsilon}-1\right)ds,\ t\in[\theta,\theta+\varepsilon),\\ 0,\ \ t\in I_1\setminus[\theta,\theta+\varepsilon),\ \ \varepsilon\in(0,\varepsilon^{**}),\end{cases} \tag{5.12}$$

$$\delta_k x(t)=\begin{cases}\int_\theta^t \lambda(\tau,t)\left[g_k[p](\tau)\delta_k p(\tau)+g_k[\tilde{p}](\tau)\delta_k p(\tau-h)\right]d\tau,\ t\in(\theta,t_1],\\ 0,\ \ t\in[t_0,\theta],\end{cases} \tag{5.13}$$

where $\lambda(\cdot)$ is the solution of the system (3.4).

By considering (5.12) in (5.13), we calculate $\delta_k x(t)$, $t\in I$. As $\theta\in I^{**}$, then by the properties of the set I^{**}, we have

$$\delta_k x(t)=\begin{cases}0, & t\in[t_0,\theta],\\ \lambda(\theta,t)g_k[p](\theta)\xi\int_\theta^t c_k(\tau)d\tau+o(\varepsilon^{k+1}),\ t\in(\theta,\theta+\varepsilon),\\ \lambda(\theta,t)g_k[p](\theta)\xi\int_\theta^{\theta+\varepsilon} c_k(\tau)d\tau+o(\varepsilon^{k+1}),\ t\in[\theta+\varepsilon,\theta+h)\cap I,\\ \lambda(\theta,t)g_k[p](\theta)\xi\int_\theta^{\theta+\varepsilon} c_k(\tau)d\tau+\chi(\theta)\lambda(\theta+h,t)g_k[\tilde{p}](\theta+h)\xi\times\\ \int_\theta^{t-h} c_k(\tau)d\tau+o(\varepsilon^{k+1}),\ t\in[\theta+h,\theta+h+\varepsilon)\cap I,\\ \left[\lambda(\theta,t)g_k[p](\theta)+\chi(\theta)\lambda(\theta+h,t)g_k[\tilde{p}](\theta+h)\right]\xi\\ \times\int_\theta^{\theta+\varepsilon} c_k(\tau)d\tau+o(\varepsilon^{k+1}),\ t\in[\theta+h+\varepsilon,t_1],\end{cases} \tag{5.14}$$

where

$$c_k(\tau)=\int_\theta^\tau \frac{(\tau-s)^{k-1}}{(k-1)!} l_k\left(\frac{2(s-\theta)}{\varepsilon}-1\right)ds,\ \tau\in[\theta,\theta+\varepsilon],\ \varepsilon\in(0,\varepsilon^{**}). \tag{5.15}$$

As $l_k(\tau)$, $\tau\in[-1,1]$ is the k-th Legendre polynomial, then it is easy to get

$$\int_{\theta}^{\theta+\varepsilon} c_k(\tau)d\tau = \frac{\varepsilon^{k+1}}{k!\,2^{k+1}}\int_{-1}^{1}(1-\tau)^k l_k(\tau)d\tau = \frac{\varepsilon^{k+1}(-1)^k}{k!\,2^{k+1}}\int_{-1}^{1}\tau^k l_k(\tau)d\tau \neq 0. \tag{5.16}$$

Taking into account (5.12)–(5.16) and the fact that $\lambda(s,t) = 0$ for $s > t$ we calculate separately each terms of (5.9). As a result, after simple reasoning we get

$$\begin{aligned}
&\delta_k x^T(t_1)\varphi_{xx}\left(x^0(t_1)\right)\delta_k x(t_1) = \xi^T\Big[g_k^T[p](\theta)\lambda^T(\theta,t_1)\varphi_{xx}\left(x^0(t_1)\right)\lambda(\theta,t_1)g_k[p](\theta)\\
&+2\chi(\theta)g_k^T[p](\theta)\lambda^T(\theta,t_1)\varphi_{xx}\left(x^0(t_1)\right)\lambda(\theta+h,t_1)g_k[\tilde{p}](\theta+h)\\
&+\chi(\theta)g^T[\tilde{p}](\theta+h)\lambda^T(\theta+h,t_1)\varphi_{xx}\left(x^0(t_1)\right)\\
&\times\lambda(\theta+h,t_1)g_k[\tilde{p}](\theta+h)\Big]\left(\xi\int_{\theta}^{\theta+\varepsilon}c_k(\tau)d\tau\right)^2+o\left(\varepsilon^{2k+2}\right),\\
&\Delta_{1k}^{*} = \xi^T\int_{\theta}^{t_1}\Big[g_k^T[p](\theta)\lambda^T(\theta,t)H_{xx}(t)\lambda(\theta,t)g_k[p](\theta)\\
&+2\chi(\theta)g_k^T[p](\theta)\lambda^T(\theta,t)H_{xx}(t)\lambda(\theta+h,t)g_k[\tilde{p}](\theta+h)\\
&+\chi(\theta)g_k^T[\tilde{p}](\theta+h)\lambda^T(\theta+h,t)H_{xx}(t)\lambda(\theta+h,t)\,g_k[\tilde{p}](\theta+h)\Big]dt\xi\left(\int_{\theta}^{\theta+\varepsilon}c_k(\tau)d\tau\right)^2+o\left(\varepsilon^{2k+2}\right),\\
&\Delta_{1k}^{**} = \frac{1}{2}\xi^T\Big[g_k^T[p](\theta)G_k[p](\theta)+2\chi(\theta)g_k^T[p](\theta)\lambda^T(\theta,\theta+h)G_k[\tilde{p}](\theta+h)\\
&+\chi(\theta)g_k^T[\tilde{p}](\theta+h)G_k[\tilde{p}](\theta+h)\Big]\xi\left(\int_{\theta}^{\theta+\varepsilon}c_k(\tau)d\tau\right)^2+o\left(\varepsilon^{2k+2}\right)
\end{aligned} \tag{5.17}$$

Substitute (5.15)–(5.17) in (5.9). Then by (3.9), (4.27), and (5.14), we have

$$\begin{aligned}
&\delta^2 S\left(u^0;\delta u\right) = -\xi^T\left\{M_k[p,p](\theta,\theta)+2\chi(\theta)M_k[p,\tilde{p}](\theta,\theta+h)\right.\\
&\left.+\chi(\theta)M_k[\tilde{p},\tilde{p}](\theta+h,\theta+h)\right\}\xi\frac{\varepsilon^{2k+2}}{(k!)^2\,4^{k+1}}\left(\int_{-1}^{1}\tau^k l_k(\tau)d\tau\right)^2\\
&+o\left(\varepsilon^{2k+2}\right),\ \forall\theta\in I^{*},\ \forall\xi\in R^{r_0}.
\end{aligned}$$

Hence, taking into account the inequality in (2.1), it is easy to complete the proof of optimality condition (5.3) for $i = k$.

Continuing the proof of Theorem 5.1, we prove also the validity of optimality condition (5.4) for $i = k$. Based on Proposition 4.2, let us consider the $(k + 1)$-th order transformation. As the equalities

http://dx.doi.org/10.5772/64225

$$L_i[p](t)+\chi(t)L_i[\tilde{p}](t+h)=0,\ P_i[p,q](t)+\chi(t)P_i[\tilde{p},\tilde{q}](t+h)=0,\ t\in I^{**},\ i=\overline{0,k},$$
$$Q_i[p](t)+\chi(t)Q_i[\tilde{p}](t+h)=0,\ t\in I^{**},\ i=\overline{0,k-1}$$

taking into account (2.6), we have

$$\begin{aligned}&\delta^2 S(u^0;\delta u)=\Delta_1^2 S(u^0;\delta_{k+1}p,\delta q,\delta_{k+1}x,\varepsilon)+\int_{\theta}^{\theta+\varepsilon}\Big[\delta_{k+1}p^T(t)\big(L_{k+1}[p](t)+L_{k+1}[\tilde{p}](t+h)\big)\\&\times\delta_{k+1}p(t)+2\delta_{k+1}p^T(t)\big(P_{k+1}[p,q](t)+P_{k+1}[\tilde{p},\tilde{q}](t+h)\big)\delta q(t)\\&+\delta_k p^T(t)\big(Q_k[p](t)+Q_k[\tilde{p}](t+h)\big)\delta_{k+1}p(t)\\&-\delta q^T(t)\big(H_{qq}(t)+H_{\tilde{q}\tilde{q}}(t+h)\big)\delta q(t)\Big]dt,\varepsilon\in(0,\varepsilon^{**}),\end{aligned}\tag{5.18}$$

where $\Delta_1^2 S\big(u^0(\cdot);\delta_{k+1}p,\delta q,\delta_{k+1}x,\varepsilon\big)$ are determined similarly to (4.33) by changing the index k by $k+1$, and $\delta_{k+1}x(t)$ is the solution of the system (similar to (4.35))

$$\begin{aligned}&\delta_{k+1}\dot{x}(t)=f_x(t)\delta_{k+1}x(t)+g_{k+1}[p](t)\delta_{k+1}p(t)+g_{k+1}[\tilde{p}](t)\delta_{k+1}p(t-h)\\&+f_q(t)\delta q(t)+f_{\tilde{q}}(t)\delta q(t-h),\ t\in[\theta,t_1],\\&\delta_{k+1}x(t)=0,\ t\in[t_0,\theta],\ \delta_{k+1}p(t)=0,\ \delta q(t)=0,\ t\in[t_0-h,\theta].\end{aligned}\tag{5.19}$$

Choose the variation $\delta u(t)=\big(\delta_0 p(t),\delta q(t)\big)^T,\quad t\in I_1$ in the following way:

$$\begin{aligned}&\delta_0 p_m(t)=0,\ t\in I_1,\ m\in\{1,2,...,r_0\}\setminus\{i,j\},\ i,j\in\{1,2,...,r_0\},\ i\neq j,\\&\delta_0 p_i(t)=\begin{cases}\alpha l_{k+1}\left(\dfrac{2(t-\theta)}{\varepsilon}-1\right),\ t\in[\theta,\theta+\varepsilon),\\0,\quad t\in I_1\setminus[\theta,\theta+\varepsilon),\end{cases}\end{aligned}\tag{5.20}$$

$$\begin{aligned}&\delta_0 p_j(t)=\begin{cases}\beta l_{k+2}\left(\dfrac{2(t-\theta)}{\varepsilon}-1\right),\ t\in[\theta,\theta+\varepsilon),\\0,\quad t\in I_1\setminus[\theta,\theta+\varepsilon),\end{cases}\\&\delta q(t)=0,\ t\in I_1,\end{aligned}$$

where $l_z(\tau)=\dfrac{1}{z!}\dfrac{1}{2^z}\dfrac{d^z}{d\tau^z}\big[\tau^2-1\big]^z$, $z\in\{k+1,k+2\}$ is a Legendre polynomials $\alpha,\beta\in R,\quad \theta\in I^{**},\ \varepsilon\in(0,\varepsilon^{**})$.

Obviously, by (5.20), the variation $\delta u(\cdot) = \left(\delta_0 p(\cdot), \delta q(\cdot)\right)^T$ defined in (5.20) satisfies conditions (4.29), (4.30) for $k+1$. Taking into account (5.20), by means of (4.30), (4.31), (4.33), and (5.19), it is easy to calculate

$$\begin{aligned} &\delta_k p(t) \sim \varepsilon^k,\ \delta_{k+1} p(t) \sim \varepsilon^{k+1},\ t \in I_1,\ \delta_{k+1} x(t) \sim \varepsilon^{k+2},\ t \in I, \\ &\Delta_1^2 S\left(u^0; \delta_{k+1} p, \delta q, \delta_{k+1} x, \varepsilon\right) \sim \varepsilon^{2k+4}. \end{aligned} \tag{5.21}$$

By (5.20) and (5.21), from (5.18) we get

$$\delta^2 S\left(u^0; \delta u\right) = \int_{\theta}^{\theta+\varepsilon} \delta_k p^T(t)\left(Q_k[p](t) + Q_k[\tilde{p}](t+h)\right)\delta_{k+1} p(t)\,dt + o\left(\varepsilon^{2k+2}\right),$$

where $Q_k[\mu](\cdot)$, $\mu \in \{p, \tilde{p}\}$ is determined in (4.25).

Hence, taking into account the skew symmetry of the matrix $Q_k[p](t) + \chi(t) Q_k[\tilde{p}](t+h)$, $t \in I$ and the properties of the set I^{**}, and also by (2.1), (4.30), and (5.20), we have

$$\begin{aligned} \delta^2 S\left(u^0; \delta u\right) &= \left[q_{ij}^{(k)}(\theta) - q_{ij}^{(k)}(\theta)\right] \int_{\theta}^{\theta+\varepsilon} \delta_k p_i(\tau)\delta_{k+1} p_j(\tau)\delta\tau + o\left(\varepsilon^{2k+2}\right) \\ &= 4(k+1)(k+2)\left(\frac{\varepsilon}{2}\right)^{2k+2} \alpha\beta ab\left[q_{ij}^{(k)}(\theta) - q_{ji}^{(k)}(\theta)\right] \int_{-1}^{1} \tau^2\left(\tau^2 - 1\right)^{2k+1} d\tau + o\left(\varepsilon^{2k+2}\right) \geq 0 \end{aligned}$$

where $\theta \in I^{**}$, $a = \dfrac{1}{(k+1)!\,2^{k+1}}$, $b = \dfrac{1}{(k+2)!\,2^{k+2}}$, and $q_{ij}^{(k)}(\theta)$, $q_{ji}^{(k)}(\theta)$ are the elements of the matrix $Q_k[p](\theta) + \chi(\theta) Q_k[\tilde{p}](\theta+h)$.

From the last inequality, by arbitrariness of $\theta \in I^{**}$, $\alpha, \beta \in R$ and $i, j \in \left\{1, 2, ..., r_0\right\}$ $(i \neq j)$ it follows that for each $\theta \in I^{**}$, the skew-symmetric matrix $Q_k[p](\theta) + \chi(\theta) Q_k[\tilde{p}](\theta+h)$ is also symmetric. Consequently, $Q_k[p](\theta) + \chi(\theta) Q_k[\tilde{p}](\theta+h) = 0$, that is, condition (5.4) is proved for *i=k*.

At last, let us prove optimality condition (5.5). Choose the variation $\delta u(t) = \left(\delta_0 p(t), \delta q(t)\right)^T$, $t \in I_1$ in the following way:

http://dx.doi.org/10.5772/64225

$$\delta_0 p(t) = \begin{cases} \xi\, l_{k+1}\left(\dfrac{2(t-\theta)}{\varepsilon} - 1\right), & t \in [\theta, \theta+\varepsilon), \\ 0, & t \in I_1 \setminus [\theta, \theta+\varepsilon), \end{cases} \tag{5.22}$$

$$\delta q(t) = \begin{cases} \eta \int_\theta^t \dfrac{(t-s)^k}{k!} l_{k+1}\left(\dfrac{2(s-\theta)}{\varepsilon} - 1\right) ds, & t \in [\theta, \theta+\varepsilon), \\ 0, & t \in I_1 \setminus [\theta, \theta+\varepsilon), \end{cases} \tag{5.23}$$

where $l_{k+1}(\tau)$, $\tau \in [-1, 1]$ is the $(1+k)$-th Legendre polynomial, $\xi \in R^{r_0}$, $\eta \in R^{r_1}$, $\theta \in I^{**}$, $\varepsilon \in (o, \varepsilon^{**})$.

Obviously, the variation $\delta u(t) = \left(\delta_0 p(t), \delta q(t)\right)^T$, $t \in I_1$ defined in (5.22) satisfies the conditions (4.29) and (4.30) for $i = 1, 2 \ldots k+1$

By (4.30), (4.31), (5.12), (5.19), (5.22), and (5.23), the following relations hold:

$$\delta_{k+1} p(t) = \begin{cases} \xi \int_\theta^t \dfrac{(t-s)^k}{k!} l_{k+1}\left(\dfrac{2(s-\theta)}{\varepsilon} - 1\right) ds, & t \in [\theta, \theta+\varepsilon), \\ 0, & t \in I_1 \setminus [\theta, \theta+\varepsilon), \end{cases} \tag{5.24}$$

$$\begin{aligned} &\delta_{k+1} p(t) \sim \varepsilon^{k+1},\ t \in I_1,\ \delta q(t) \sim \varepsilon^{k+1},\ t \in I_1,\ \delta_{k+1} x(t) \sim \varepsilon^{k+2},\ t \in I, \\ &\Delta_1^2 S\left(u^0; \delta_{k+1} p, \delta q, \delta_{k+1} x, \varepsilon\right) \sim \varepsilon^{2k+4}. \end{aligned} \tag{5.25}$$

Taking into account (5.23)–(5.25) and validity of the equality $Q_k[p](t) + \chi(t) Q_k[\tilde{p}](t+h) = 0$, $t \in I^{**}$ (see (5.4)), from (5.18), we get

$$\begin{aligned} \delta^2 S\left(u^0; \delta u\right) = \left(\frac{\varepsilon}{2}\right)^{2k+3} \Big[& \xi^T \left(L_{k+1}[p](\theta) + \chi(\theta) L_{k+1}[\tilde{p}](\theta+h)\right) \xi \\ & + 2\xi^T \left(P_{k+1}[p, q](\theta) + \chi(\theta) P_{k+1}[\tilde{p}, \tilde{q}](\theta+h)\right) \eta \\ & - \eta^T \left(H_{qq}(\theta) + \chi(\theta) H_{\tilde{q}\tilde{q}}(\theta+h)\right) \eta \Big] \frac{1}{(k!)^2} \int_{-1}^{1} \left(\int_{-1}^{t} (t-s)^k\, p_{k+1}(s)\, ds \right)^2 dt + o\left(\varepsilon^{2k+3}\right). \end{aligned}$$

From this expansion, taking into account (2.1), it follows inequality (5.5).

Therefore, Theorem 5.1 is completely proved.

Corollary 5.1. Let all the conditions of Theorem 5.1 be fulfilled. Let, in addition, the following equalities hold:

$$L_i[p](t) + \chi(t) L_i[\tilde{p}](t+h) = 0, \ \forall t \in I^{**}, \ i = 0,1,...$$

Then, for optimality of the singular control $u^0(\cdot)$, it is necessary that the relations

$$\begin{aligned}
&P_i[p,q](\theta) + \chi(\theta) P_i[\tilde{p},\tilde{q}](\theta+h) = 0, \ i = 0,1,...; \\
&\xi^T \{M_i[p,p](\theta,\theta) + 2\chi(\theta) M_i[p,\tilde{p}](\theta,\theta+h) \\
&+ \chi(\theta) M_i[\tilde{p},\tilde{p}](\theta+h,\theta+h)\} \xi \le 0, \ i = 0,1,...; \\
&Q_i[p](\theta) + \chi(\theta) Q_i[\tilde{p}](\theta+h) = 0, \ i = 0,1,...
\end{aligned}$$

be fulfilled for all $\theta \in I^{**}$, $\xi \in R^{r_0}$.

The proof of the corollary follows immediately from Theorem 5.1.

Remark 5.1. As is seen (see Proposition 3.1 and (4.6), (4.15), and (4.24)), for validity of optimality conditions (5.2)–(5.4), for $k = 0$ it is sufficient that assumptions (A1) and (A2) be fulfilled.

Remark 5.2. It is clear that (see Proposition 4.1) for validity of optimality conditions (5.5), for $k = 0$ it is sufficient that assumptions (A1), (A3), and (A4) be fulfilled.

Remark 5.3. If in Definition 2.1 a special plot is some interval $(\bar{t}, \hat{t}) \subset I$, then very easily similar to the proof of Theorem (5.1) we can prove that conditions (5.2)–(5.5) as optimality conditions are valid for all $\theta \in (\bar{t}, \hat{t}) \cap I^{**}$ and $\xi \in R^{r_0}$, $\eta \in R^{r_1}$.

6. Conclusion

As is seen, systems (1.2) and (1.3) are not the most general among all the systems with retarded control. We have chosen it only for definiteness, just to demonstrate the essentials of our method. Nevertheless, the optimality conditions (5.2)–(5.5) can be generalized to the case for more general systems with retarded control.

It should be noted that (1) optimality conditions (5.4) and (5.5), for $k = 0$, are actually the analogs of the equality-type conditions and the Kelly [12] condition, while optimality condition (5.3) is the analog of the Gabasov [11] condition for the considered problem (1.1)–(1.3); (2) optimality condition (5.5), for $k = 1$ is the analog of the Koppa-Mayer [33] condition. Conditions (5.3)–(5.5) were obtained in [10] only for singular controls with complete degree of degeneracy, that is, for the case when $r_1 = 0$ (see Definition 2.1).

http://dx.doi.org/10.5772/64225

We also note that (1) the analog of the Kelly condition and equality-type condition was obtained in [24] by another method for systems with retarded state; (2) optimality-type conditions (5.2)–(5.5) for system with retarded state were obtained in [[31, 32], p. 119]; (3) optimality conditions of type (5.4), (5.5) for systems without retardation were obtained in the papers [[23, 26, 27], p. 145, [29, 30, 33, 34, 39–41], etc.].

The proof of Theorem 5.1 shows that the optimality conditions (5.3)–(5.5) are independent. Also, it is clear that, unlike (5.2), (5.3), and (5.5), the optimality condition (5.4) for $r_1 = r - 1$ (see Definition 2.1) becomes ineffective, though it is effective in the general case for $r_1 < r - 1$. To illustrate the rich content of condition (5.4), we consider a concrete example:

Example. $\dot{x}_1(t) = u_2(t) + u_1^2(t-1) - u_3(t-1)$, $\dot{x}_2(t) = u_1(t) - u_2(t)$,

$\dot{x}_3(t) = \left(u_1(t) + u_2(t)\right)\ x_2(t) + u_3^2(t) + u_3^2(t-1),\ t \in I := [0,2],\ x_i(0) = 0,\ u_i(t) = 0,\ t \in [-1,0),$

$\left|u_i\right| < 2,\ i = 1,2,3, h = 1,\ \phi(x(2)) = x_3(2) + \frac{1}{2}x_1^2(2) \to \min.$

Check for optimality of the control $u^0(t) = (0,0,0)^T,\ t \in [-1,2]$. In this control according to (2.7), (2.8), (3.9), (3.10), (4.6), (4.9), (4.15), (4.21), and (4.24), we have

$x_i^0(t) = 0,\ i = 1,2,3,\ \psi_i^0(t) = 0, i = 1,2,\ \psi_3^0(t) = -1,\ t \in I,$

$H\left(\psi^0(t), x, u, v, t\right) = -\left(u_1 + u_2\right)x_2 - u_3^2 - v_3^2,\quad H_{uu}(t) := \left(h_{ij}(t)\right), t \in I,\quad \text{where}\quad h_{ij}(t) = 0,$

$i, j \in \{1,2,3\},\quad (i,j) \neq (3,3),\quad h_{33}(t) = -2; H_{vv}(t+1) = \left(\tilde{h}_{ij}(t)\right),\ t \in [0,1],\quad \text{where}$

$\tilde{h}_{ij}(t) = 0, i, j \in \{1,2,3\},\quad (i,j) \neq (3,3),\quad \tilde{h}_{33}(t) = -2;\quad H_{vv}(t+1) = 0, t \in (1,2];$

$g_0^T[p](t) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -1 & 0 \end{pmatrix}, t \in I,\quad g_0^T[\tilde{p}](t) = 0,\quad t \in I,\quad \text{where}\quad p := \left(u_1, u_2\right),\quad \tilde{p} := \left(v_1, v_2\right);$

$g_1[p](t) = g_1[\tilde{p}](t) = 0,\ t \in I,\quad Q_0[p](t) = \begin{pmatrix} 0 & -2 \\ 2 & 0 \end{pmatrix},\quad t \in I,\quad Q_0[\tilde{p}](t+1) = 0,\quad t \in I,$

$L_1[p](t) = L_1[\tilde{p}](t+1) = 0,\quad t \in I,\quad P_0[p,q](t) = P_0[\tilde{p},\tilde{q}](t) = 0,\quad P_1[p,q](t) = P_1[\tilde{p},\tilde{q}](t) = 0,$

$M_0[p,p](t,t) = \begin{pmatrix} -1 & -1 \\ 1 & 0 \end{pmatrix}, t \in I,\quad M_0[p,\tilde{p}](\cdot) = 0,\quad M_0[\tilde{p},\tilde{p}](\cdot) = 0,$

$H_{qq}(t) + H_{\tilde{q}\tilde{q}}(t+1) = \begin{cases} -4, & t \in [0,1), \\ -2, & t \in [1,2], \end{cases}$ where $q = u_3,\ \tilde{q} = v_3.$

Hence, we have the following: (1) admissible control $u^0(t) = (0,0,0)^T,\ t \in [-1,2]$ is singular (in the sense of Definition 2.1) and singularity to it is delivered by the vector component $p = \left(u_1, u_2\right)^T$, that is, equality (5.1) is fulfilled only $k = 0$; (2) optimality conditions (5.2), (5.3), (5.5), and the results of the papers [1–3, 6, 9, 10] cannot say that whether the control $u^0(\cdot)$ is an optimal or not. However, optimality condition (5.4) for $k = 0$ is not fulfilled

$(Q_0[p](t) + \chi(t)Q_0[\tilde{p}](t+1) = \begin{pmatrix} 0 & -2 \\ 2 & 0 \end{pmatrix} = 0,\ t \in I)$, that is, by condition (5.4) (for $k = 0$) we conclude that the control $u^0(t) = (0,0,0)^T,\ t \in [-1,2]$ cannot be optimal.

Author details

Misir J. Mardanov[1*] and Telman K. Melikov[2]

*Address all correspondence to: misirmardanov@imm.az

1 Institute of Mathematics and Mechanics of ANAS, Baku, Azerbaijan

2 Institute of Mathematics and Mechanics and Institute of Control Systems of ANAS, Baku, Azerbaijan

References

[1] Artyunov A, Mardanov M: To theory of maximum principle in delay problems. Differen. Uravn. 1989; 25: 2048–2058.

[2] Halanay A: Optimal controls for systems with time lag. SIAM J. Control, 1968; 6(2). DOI: 10.1137/030606016

[3] Kharatashvili G, Tadumadze T: Nonlinear optimal control systems with variable constructions. Matem. Sb. 1978; 107(149): No 4 (12), 613–633.

[4] Mardanov M: Necessary conditions of optimality in delay and phase restraint systems. Matem. zhametki, 1987; 42(5), 691–702.

[5] Mardanov M: Investigation of optimal delay process with constraints. Elm, Baku; 2009. 192 p.

[6] Matveev A: Optimal control problems with general form delays and phase constraints. Izv. AN SSSR. ser. matem. 5(6):1200–1229. DOI: 10.1070/IM1989v033n03ABEH000855

[7] Tadumadze T: Some problems of quality theory of optimal control. Tbilisi, Izd. TTU; 1983. 128 p.

[8] Vasil'ev F: Conditions of optimality for some classes of systems unsolved with regard to derivative. DAN SSSR, 1969; 184(6), 1267–1270.

[9] Guliyev V: Some issues of optimal control theory in delay argument systems involving constraints. Author's thesis for cand. of phys. math. sci. Baku, 1985. 25 p.

http://dx.doi.org/10.5772/64225

[10] Mardanov M, Melikov T: Recurrent optimality conditions of singular controls in delay control systems. The Third International Conference "Problems of Cybernetics and Informatics" 6–8 September 2010; Baku, pp. 3–5.

[11] Gabasov R: To theory of necessary conditions of optimality for singular controls. DAN SSSR, 1968; 183(2), 300–302.

[12] Kelley H: Necessary conditions for singular extremals based on the second variation. Raketnaya tekhnika i kosmonavtika, 1964; (8), 26–29.

[13] Kelley H, Kopp R, Moyer H: Singular extremals, in "Topics in Optimization" (ed. By Leitman G), Acad. Press, New York-London, 1967; 63–101.

[14] Akhmedov K, Melikov T, Gasanov K: On optimality of singular controls in delay systems. Dokl. AN Az SSR, 1975, 31(7), 7–10.

[15] Mansimov K: Multipoint necessary condition of optimality for singular in the classic sence controls in delay control systems. Diff. Uravn. 1985; 21(3), 527–530.

[16] Mardanov M: On condition of optimality for singular controls. DAN SSSR, 1980; 253(4), 815–818.

[17] Melikov T: Recurrent conditions of optimality for singular controls in delay systems. Dokl. RAN, 1992; 322(5), 843–846.

[18] Melikov T: An Analogue of the Kelley condition for optimal systems with aftereffect of neutral type. Zhurnal Vychisl. Mat. i Mat. Fiz., 1998; 38(9), 1490–1499.

[19] Melikov T: Optimality of singular controls in systems with aftereffect of neutral type. Zhurnal Vychisl. Mat. i Mat. Fiz., 2001; 41(9), 1332–1343.

[20] Parayev Y.I. Solving a problem of optimal product, storage and sale of goods. Izv. RAN. Teor. i sistemy upravlenia. 2000; (2), 103–107.

[21] Sansone J: Ordinary differential equations. Inostrannaya literatura. Moscow, Il, 1954; 2.

[22] Rozonoer L: L.S. Pontryagin's maximum principle in theory of optimal systems, III. Avtomatika i telemekhanika, 1959; 20(12), 1561–1578.

[23] Agrachev A, Gamkrelidze R: Second order optimality principle for contagion problem. Matem. sb. 1976; 100(142), No 4(8), 610–643.

[24] Ashepkov L, Eppel D: Analog of Kelley condition in optimal delay systems. Diff. Uravn., 1974; 10(4), 591–597.

[25] Barbashina E: Kopp-Moyer type necessary conditions for Goursat-Darboux systems. 1989; 25(6), 1045–1047.

[26] Bolonkin A: Special extremals in optimal control problems. Izv. AN SSSR. OTN. Tekhnicheskaya kibernetika, 1969; 2, 187–198.

[27] Gabasov R, Kirillova F: Singular optimal controls. Nauka, Moscow; 1973. 256 p.

[28] Gasanov K, Yusifov B: Inductive analysis of singular controls in delay systems. Avtomatika i telemekhanika, 1982; (6), 37–42.

[29] Goh B: Necessary conditions for singular extremals involving multiple control variables. SIAM J. Control, 1966; 4(4), 716–731.

[30] Gorokhovik V, Gorokhovik S: Different forms of Legendre-Clebsch generalized conditions. Avtomatika i telemekhanika, 1982; (7), 28–33.

[31] Jacobson D: A new necessary condition of optimality for singular control problems. SIAM J. Control, 1969; 7(4), 578–595. DOI: 10.1137/0307042.

[32] Kaganovich S: On inductive method for studying singular extremals. Avtomatika i telemekhanika, 1976; (11), 28–39.

[33] Kopp R, Moyer H: Necessary condition of optimality for singular extremals. Raketnaya tekhnika i kosmonavtika. 1965; 3(8), 84–91.

[34] Krener A: The high order maximal principle and its application to singular extremals. SIAM J. Control Optimization, 1977; 15(2), 256–293.

[35] Mardanov M: Second order necessary condition of optimality in delay control systems. UMN, 1988; 43, 4(262), 213–214.

[36] Melikov T: The necessary conditions for high-order optimality. Zhurnal Vychisl. Mat. i Mat. Fiz., 1995; 35(7), 1134–1138.

[37] Melikov T: Singular controls in aftereffect systems. Baku, "Elm" 2002; 188 p.

[38] Melikov T: Recurrent conditions of optimality for singular controls in Goursat-Darboux systems. Dokl. Akad. Nauk. Az. SSR, 1990; 14(8), 6–10.

[39] Srochko V: Investigation of second variation on singular controls. Diff. Uravn., 1974; 10(6), 1050–1066.

[40] Srochko V: Method of variation transformation in theory of singular controls. Differen. i integ. uravnenia., Irkutsk: State Univ., 1973; (2), 70–80.

[41] Vopnyarskiy I: Theorems of the existence of optimal control in Bolts problem, some its applications and necessary conditions of optimality of sliding and singular modes. Zhurn. Vych. math. i math-phys., 1967; 7(2), 259–283.

[42] Kudryavtsev L: Course of mathematical analysis. Vyshaya shkola, Moscow; 1981; (2), 2.

[43] Mardanov M: Legendre's necessary conditions in delay control optimization problems. DAN SSSR, 1987. 297(4), 795–797.

Chapter 9

Simultaneous H^∞ Control for a Collection of Nonlinear Systems in Strict-Feedback Form

Jenq-Lang Wu , Chee-Fai Yung and Tsu-Tian Lee

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64105

Abstract

Based on the control storage function approach, a constructive method for designing simultaneous H^∞ controllers for a collection of nonlinear control systems in strict-feedback form is developed. It is shown that under mild assumptions, common control storage functions (CSFs) for nonlinear systems in strict-feedback form can be constructed systematically. Based on the obtained common CSFs, an explicit formula for constructing simultaneous H^∞ controllers is presented. Finally, an illustrative example is provided to verify the obtained theoretical results.

Keywords: nonlinear control systems, simultaneous H^∞ control, state feedback, storage functions, strict-feedback form

1. Introduction

The simultaneous H^∞ control problem concerns with designing a single controller which simultaneously renders a set of systems being internally stable and satisfying an L_2-gain specification. In the last decades, there have been some researchers studying the simultaneous H^∞ control problem in linear case, see references [1–6]. In references [1] and [2], necessary and sufficient conditions for the simultaneous H^∞ control via nonlinear digital output feedback controllers were derived by using the dynamic programming approach. In reference [3], a numerical design method was proposed for designing simultaneous H^∞ controllers. In reference [4], it was shown that the simultaneous H^∞ control problem is equivalent to a strong H^∞ control problem. In reference [5], linear periodically time-varying controllers were employed for simultaneous H^∞ control. In reference [6], a simultaneous H^∞ control problem was solved via the chain scattering framework.

All the results mentioned earlier are derived for linear systems case. Till now, only few results have been reported about simultaneous H^∞ control of nonlinear systems, see references [7, 8]. In reference [7], a control storage function (CSF) method was developed for designing simultaneous H^∞ state feedback controllers for a collection of single-input nonlinear systems. Necessary and sufficient conditions for the existence of simultaneous H^∞ controllers were derived. Moreover, an explicit formula for constructing simultaneous H^∞ feedback controllers was proposed. The CSF approach was first introduced in reference [9]. It is motivated by the control Lyapunov function (CLF) method (please see references [10–18]) for designing stabilizing controllers of nonlinear control systems. One difficulty in applying CSFs/CLFs for solving control problems is that how to derive CSFs/CLFs for nonlinear systems is an open problem unless they are in some particular forms. No systematic methods for constructing CSFs have been proposed in reference [7]. It is important to identify those nonlinear systems whose corresponding CSFs/CLFs exist and can be constructed systematically. In reference [8], the CSF approach was applied to design simultaneous H^∞ controllers for a collection of nonlinear control systems in canonical form. It was shown that under mild assumptions, CSFs can be constructed systematically for nonlinear systems in canonical form; and simultaneous H^∞ control for such systems can be easily achieved. In this chapter, we further study the simultaneous H^∞ control problem for nonlinear systems in strict-feedback form. It is known that the strict-feedback form is more general than the canonical form. Moreover, a restrictive assumption made in reference [8] is relaxed in this chapter. Based on the CSF approach and by using the backstepping technique, we develop a systematic method for constructing simultaneous H^∞ state feedback controllers. The proposed results in reference [8] are special cases of the results presented in this chapter.

2. Problem formulation and preliminaries

In this section, the simultaneous H^∞ control problem to be solved will be formulated and some preliminaries will be presented. For simplifying the expressions, we use the same notations x, u, w, and z to denote the states, control inputs, exogenous inputs, and the controlled outputs of all the considered systems.

2.1. Problem formulation

Consider a collection of nonlinear control systems:

$$\begin{aligned} \dot{x} &= f_i(x) + g_{1i}(x)w + g_{2i}(x)u \\ z &= h_{1i}(x) + k_{11i}(x)w, \quad \mathrm{i} = 1,\ldots,q, \end{aligned} \tag{1}$$

where $x = \left[x_1, x, \cdots, x_n\right]^T \in R^n$ is the state, $w \in R^m$ is the disturbance input, $u \in R$ is the control input, $z \in R^r$ is the controlled output, $f_i : R^n \mapsto R^n, g_{1i} : R^n \mapsto R^{n \times m}, g_{2i} : R^n \mapsto R^n, h_{1i} : R^n \mapsto R^r,$

http://dx.doi.org/10.5772/64105

and $k_{11i}: R^n \mapsto R^{r\times m}, i = 1, ..., q$, are smooth functions. Here we denote the i-th system in Eq. (1) as system S_i. For all i=1,…,q, suppose that $f_i(0) = 0$ and $h_{1i}(0) = 0$. For convenience, define $\bar{x}_j = [x_1, x_2, \cdots, x_j]^T \in R^j, j = 1, ..., n$. Suppose that $f_i(x)$, $g_{1i}(x)$, and $g_{2i}(x)$, i=1,…,q, have the following forms:

$$f_i(x) = \begin{bmatrix} x_2 + \theta_{i1}(x_1) \\ \vdots \\ x_{j+1} + \theta_{ij}(\bar{x}_j) \\ \vdots \\ x_n + \theta_{i(n-1)}(\bar{x}_{n-1}) \\ \theta_{in}(x) \end{bmatrix}, \quad g_{1i}(x) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \rho_i(x) \end{bmatrix}, \quad g_{2i}(x) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \eta_i(x) \end{bmatrix}. \tag{2}$$

where $\theta_{ij}: R^j \mapsto R, \rho_i: R^n \mapsto R^{1\times m}$, and $\eta_i: R^n \mapsto R, i = 1, 2, ..., q, j = 1, ..., n$, are smooth functions with $\theta_{ij}(0) = 0$ and $\eta_i(x) \neq 0$ for each $x \in R^n$. Assume that all functions $\eta_i(x), i = 1, ..., q$, have the same sign. Without loss of generality, suppose that $\eta_i(x) > 0, i = 1, ..., q$. By Eq. (2), the q possible models can be explicitly expressed as

$$\begin{aligned} \dot{x}_1 &= x_2 + \theta_{i1}(x_1) \\ &\vdots \\ \dot{x}_j &= x_{j+1} + \theta_{ij}(\bar{x}_j) \\ &\vdots \\ \dot{x}_{n-1} &= x_n + \theta_{i(n-1)}(\bar{x}_{n-1}) \\ \dot{x}_n &= \theta_{in}(x) + \rho_i(x)w + \eta_i(x)u, \\ z &= h_{1i}(x) + k_{11i}(x)w, \qquad i = 1, 2, .., q. \end{aligned} \tag{3}$$

Suppose that the following assumption holds.

Assumption 1: $\gamma^2 I - k_{11i}^T(x)k_{11i}(x) > 0$. $\forall x \in R^n$ and $\forall i \in \{1, ..., q\}$.

It is clear that we can always find a positive (semi)definite function $U(x)$ such that, for all $i \in \{1,…q\}$,

$$h_{1i}^T(x)h_{1i}(x) + h_{1i}^T(x)k_{11i}(x)\left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)^{-1} k_{11i}^T(x)h_{1i}(x) \le U(x), \ \forall x \in R^n.$$

The objective of this chapter is to find a continuous function $p: R^n \mapsto R$ such that the state feedback controller

$$u = p(x) \tag{4}$$

internally stabilizes the systems in Eq. (3) simultaneously; and, for each $T > 0$ and for each $w_i \in L_2[0, T]$, all closed-loop systems, starting from the initial state $x(0) = 0$, satisfy (for a given $\gamma > 0$)

$$\int_0^T z^T(t)z(t)dt \le \hat{\gamma}^2 \int_0^T w^T(t)w(t)dt \quad \text{for some } \hat{\gamma} < \gamma. \tag{5}$$

2.2. Control storage functions

Here we review some important concepts about the CSF method introduced in references [7, 9].

Definition 1 [7, 9]: Consider the system S_i in Eq. (1). A smooth, proper, and positive definite function $V_i: R^n \to R$ is a CSF of S_i if, for each $x \in R^n \backslash \{0\}$ and each $w \in R^m$,

$$\left[\inf_{u \in R} \left\{ \frac{\partial V_i(x)}{\partial x}\left(f_i(x) + g_{1i}(x)w + g_{2i}(x)u \right) + \left(h_{1i}(x) + k_{11i}(x)w \right)^T \left(h_{1i}(x) + k_{11i}(x)w \right) - \gamma^2 w^T w \right\} < 0. \right]$$

For ensuring the continuity of the obtained simultaneous H^∞ controllers, the *L_2-gain small control property* (*L_2-gain SCP*) has been introduced in reference [7].

Definition 2 [7]: A CSF $V_i: R^n \to R$ of S_i satisfies the L_2-gain SCP if for each $\varepsilon > 0$, there is a $\delta_1 > 0$ and a $\delta_2 > 0$ such that, if $x \neq 0$ satisfies $\|x\| < \delta_1$ and w satisfies $\|w\| < \delta_2$, there is some u with $|u| < \varepsilon$ satisfying

$$\frac{\partial V_i(x)}{\partial x}\left(f_i(x) + g_{1i}(x)w + g_{2i}(x)u \right) + \left(h_{1i}(x) + k_{11i}(x)w \right)^T \left(h_{1i}(x) + k_{11i}(x)w \right) - \gamma^2 w^T w < 0.$$

3. Main results

For a single system, it has been shown in reference [7] that the existence of CSFs is a necessary and sufficient condition for the existence of H^∞ controllers. Therefore, for the existence of simultaneous H^∞ controllers for the systems in Eq. (3), the existence of CSFs for these systems is necessary. In references [7] and [9], no systematic methods have been proposed for constructing CSFs. Here, based on the backstepping method, we first derive CSFs explicitly for the systems in Eq. (3).

Let

http://dx.doi.org/10.5772/64105

$$s_1(x_1) = x_1,$$

$$\hat{V}_1(x_1) = \frac{1}{2}s_1^2(x_1).$$

It is easy to show that we can find a function $\varphi_1 : R \to R$ and a positive definite function $\mu_1 : R \to R$ such that

$$x_1 \cdot \left(\varphi_1(x_1) + \theta_{i1}(x_1)\right) \le -\mu_1(x_1), \text{ for all } i = 1,\ldots,q.$$

For j=2,…,n, let

$$s_j(\bar{x}_j) = x_j - \varphi_{j-1}(\bar{x}_{j-1}),$$

$$\hat{V}_j(\bar{x}_j) = \hat{V}_{j-1}(\bar{x}_{j-1}) + \frac{1}{2}s_j^2(\bar{x}_j).$$

Similarly, we can find functions $\varphi_j : R^j \to R, j = 2, ..., n-1$, and positive definite function $\mu_j : R^j \to R, j = 2, ..., n-1$, such that

$$\sum_{l=1}^{j-1} \frac{\partial \hat{V}_j(\bar{x}_j)}{\partial x_l}\left(x_{l+1} + \theta_{il}(\bar{x}_l)\right) + \frac{\partial \hat{V}_j(\bar{x}_j)}{\partial x_j}\left(\varphi_j(\bar{x}_j) + \theta_{ij}(\bar{x}_j)\right) \le -\sum_{l=1}^{j} \mu_l(\bar{x}_l), \quad \text{for all } i = 1,\ldots,p.$$

Then, it is clear that the function

$$\hat{V}(x) \equiv \frac{1}{2}\sum_{j=1}^{n} s_j^2(\bar{x}_j)$$

is positive definite, and radially unbounded.

Now, we discuss the existence of common CSFs for the systems in Eq. (3). For convenience, we say that a continuous function $v(\bar{x}_j)$ is dominated by a continuous function $v(\bar{x}_j)$ if there exists a constant $c > 0$ such that $v(\bar{x}_j) < cv(\bar{x}_j)$ for all $\bar{x}_j \neq 0$.

***Theorem 1*:** Consider the systems in Eq. (3). Suppose that *Assumption 1* holds. If the functions $\varphi_j: R^j \to R$, j=1,…,n-1, are such that $U(x)\Big|_{s_n(x)=0}$ is dominated by $\sum_{l=1}^{n-1}\mu_l(\bar{x}_l)$, then there exists a common *CSF* that satisfies the L_2-*gain SCP* for all the systems in Eq. (3).

Proof: Let $V(x) = K \cdot \hat{V}(x)$, where $K > 0$ will be specified later. For system S_i, define the corresponding Hamiltonian function as

$$H_i(x,w,u) \equiv \dot{V}(x) + \left(h_{1i}(x) + k_{11i}(x)w\right)^T \left(h_{1i}(x) + k_{11i}(x)w\right) - \gamma^2 w^T w.$$

By the backstepping method, we can show that

$$\begin{aligned}
&H_i(x,w,u)\\
&= K\sum_{j=1}^{n} s_j(\bar{x}_j)\cdot \dot{s}_j(\bar{x}_j) + \left(h_{1i}(x) + k_{11i}(x)w\right)^T \left(h_{1i}(x) + k_{11i}(x)w\right) - \gamma^2 w^T w\\
&\le -K\sum_{i=1}^{n-1}\mu_j(\bar{x}_j) \; + Ks_n(x)\left(s_{n-1}(\bar{x}_{n-1}) + \theta_{in}(x) + \rho_i(x)w + \eta_i(x)u - \sum_{l=1}^{n-1}\frac{\partial \phi_{n-1}(\bar{x}_{n-1})}{\partial x_l}\cdot\left(x_{l+1} + \theta_{il}(\bar{x}_l)\right)\right)\\
&+ h_{1i}^T(x)h_{1i}(x) + h_{1i}^T(x)k_{11i}(x)w + w^T k_{11i}^T(x)h_{1i}(x) - w^T\left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)w.
\end{aligned}$$

After some manipulations, we have

$$\begin{aligned}
H_i(x,w,u) &\le a_i(x) + b_i(x)\cdot u - (w - w_{i*}(x))^T\left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)(w - w_{i*}(x))\\
&\le a_i(x) + b_i(x)\cdot u,
\end{aligned} \tag{6}$$

where

$$\begin{aligned}
a_i(x) &= -K\sum_{j=1}^{n-1}\mu_j(\bar{x}_j) + Ks_n(x)\left(s_{n-1}(\bar{x}_{n-1}) + \theta_{in}(x) - \sum_{l=1}^{n-1}\frac{\partial \varphi_{n-1}(\bar{x}_{n-1})}{\partial x_l}\cdot\left(x_{l+1} + \theta_{il}(\bar{x}_l)\right)\right) + h_{1i}^T(x)h_{1i}(x)\\
&+\left(\frac{K}{2}s_n(x)\rho_i^T(x) + k_{11i}^T(x)h_{1i}(x)\right)^T\left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)^{-1}\left(\frac{K}{2}s_n(x)\rho_i^T(x) + k_{11i}^T(x)h_{1i}(x)\right)\\
b_i(x) &= K\eta_i(x)s_n(x)\\
w_{i*}(x) &= \left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)^{-1}\left(\frac{K}{2}s_n(x)\rho_i^T(x) + k_{11i}^T(x)h_{1i}(x)\right)
\end{aligned}$$

Therefore, $V(x) = K \cdot \hat{V}(x)$ is a CSF of S_i if

http://dx.doi.org/10.5772/64105

$$\forall x \neq 0 \text{ such that } b_i(x) = 0 \Rightarrow a_i(x) < 0.$$

As $U(x)\Big|_{s_n(x)=0}$ is dominated by $\sum_{l=1}^{n-1} \mu_l(\bar{x}_l)$, we can choose a $K > 0$ such that

$$U(x)\Big|_{s_n(x)=0 \text{ and } x\neq 0} - K\sum_{j=1}^{n-1} \mu_j(\bar{x}_j) < 0.$$

Notice that $b_i(x) = 0$ if and only if $s_n(x) = 0$. Therefore,

$$a_i(x)\,|_{b_i(x)=0 \text{ and } x\neq 0} \leq -K\sum_{j=1}^{n-1} \mu_j(\bar{x}_j) + U(x)\,|_{s_n(x)=0 \text{ and } x\neq 0} < 0. \tag{7}$$

This shows that $V(x)$ is a CSF for the i-th system in Eq. (3). Since Eq. (7) holds for all $i \in \{1,\ldots,q\}$, $V(x)$ is a common CSF for all the systems in Eq. (3).

Now we prove that $V(x)$ satisfies the L_2-gain SCP. Note that if we can find a continuous stabilizing feedback law $d_i(x)$ with $d_i(0) = 0$ such that $H_i(x, w, d_i(x)) < 0$ for each $x \in R^n \backslash \{0\}$ and each $w \in R^m$, then $V(x)$ satisfies the L_2-gain SCP. Let

$$d_i(x) = -\frac{1}{\eta_i(x)}\Bigg(s_{n-1}(\bar{x}_{n-1}) + \theta_{in}(x) - \sum_{l=1}^{n-1} \frac{\partial \varphi_{n-1}(\bar{x}_{n-1})}{\partial x_l} \cdot \left(x_{l+1} + \theta_{il}(\bar{x}_l)\right) + \rho_i(x)\left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)^{-1}\left(\frac{K}{4} s_n(x)\rho_i^T(x) + k_{11i}^T(x)h_{1i}(x)\right)\Bigg) - \hat{\mu}_n(x)$$

where the continuous function $\hat{\mu}_n(x)$ with $\hat{\mu}_n(0) = 0$ is such that $s_n(x)\hat{\mu}_n(x) > 0$ if $s_n(x) \neq 0$, and

$$-K\sum_{j=1}^{n-1} \mu_j(\bar{x}_j) - K\eta_i(x)s_n(x)\hat{\mu}_n(x) + U(x) < 0 \quad \forall x \neq 0.$$

Note that such $\hat{\mu}_n(x)$ always exists since $U(x)\Big|_{s_n(x)=0}$ is dominated by $\sum_{l=1}^{n-1} \mu_l(\bar{x}_l)$. Clearly, $d_i(x)$ is continuous in R^n and $d_i(0) = 0$. By Eq. (6), we have

$$
\begin{aligned}
H_i(x,w,d_i(x)) &\le a_i(x)+b_i(x)d_i(x)\\
&= -K\sum_{j=1}^{n-1}\mu_j(\bar{x}_j)-K\eta_i(x)s_n(x)\hat{\mu}_n(x)\\
&\quad + h_{1i}^T(x)h_{1i}(x)+h_{1i}^T(x)k_{11i}(x)\left(\gamma_i^2 I-k_{11i}^T(x)k_{11i}(x)\right)^{-1}k_{11i}^T(x)h_{1i}(x)\\
&\le -K\sum_{j=1}^{n-1}\mu_j(\bar{x}_j)-K\eta_i(x)s_n(x)\hat{\mu}_n(x)+U(x)<0,\ \ \forall x\neq 0,\forall w.
\end{aligned}
$$

This implies that $V(x)$ satisfies the L_2-gain SCP and completes the proof.

To derive simultaneous H^∞ controllers, define (for i=1,…,q)

$$
p_i(x)\equiv\begin{cases}-\dfrac{a_i(x)+\sqrt{a_i^2(x)+\beta_i b_i^4(x)}}{b_i(x)}, & \text{if } s_n(x)\neq 0\\ 0, & \text{if } s_n(x)=0\end{cases}
$$

where $\beta_i > 0$, i=1,…,q, are given constants. Since $V(x)$ satisfies the L_2-gain SCP, the functions $p_i(x)$, i=1,…,q, are continuous in R^n [16]. We have the following results.

Theorem 2: Consider the collection of systems in Eq. (3). Suppose that *Assumption 1* holds. If the functions $\varphi_j: R^j \to R, j=1,...,n-1$, are such that $U(x)\Big|_{s_n(x)=0}$ is dominated by $\sum_{l=1}^{n-1}\mu_l(\bar{x}_l)$, then a continuous function $p: R^n \to R$ exists such that the feedback law defined in Eq. (4) internally stabilizes the collection of systems in Eq. (3) simultaneously; and moreover, all the closed-loop systems satisfy the L_2-gain requirement specified in Eq. (5). In this case,

$$
u=p(x)\equiv\begin{cases}\min\limits_{i\in\{1,2,...,q\}}\{p_i(x)\}, & \text{if } s_n(x)>0\\ 0, & \text{if } s_n(x)=0\\ \max\limits_{i\in\{1,2,...,q\}}\{p_i(x)\}, & \text{if } s_n(x)<0\end{cases} \tag{8}
$$

is a simultaneous H^∞ controller for all the systems in Eq. (3).

Proof: Since the functions $p_i(x)$, i=1,2,…,q, are continuous in R^n, from the definition of $p(x)$, its continuity is obvious. In the following, we first prove the achievement of L_2-gain requirement [Eq. (5)], and then the internal stability of all the closed-loop systems.

A. L_2-gain requirement

Since $H_i(x,w,u) \le a_i(x)+b_i(x)u$, if we can show that

http://dx.doi.org/10.5772/64105

$$a_i(x) + b_i(x)p(x) < 0 \quad \forall x \neq 0, i = 1,\ldots,q, \tag{9}$$

Then, with the controller defined in Eq. (8), all the closed-loop systems satisfy the L_2-gain requirement specified in Eq. (5).

1. $s_n(x) = 0$ and $x \neq 0$.

In this case, $u = p(x) = 0$ and $b_i(x) = 0$. Then, by Eq. (7),

$$a_i(x) + b_i(x) \cdot p(x) = a_i(x) < 0, \quad i = 1,\ldots,q.$$

2. $s_n(x) > 0$.

In this case, since $b_i(x) > 0$, we have

$$\begin{aligned} a_i(x) + b_i(x)p(x) &= a_i(x) + b_i(x) \cdot \min_{j\in\{1,2,\ldots,q\}} \{p_j(x)\} \\ &\leq a_i(x) + b_i(x) \cdot p_i(x) \\ &= -\sqrt{a_i^2(x) + \beta_i b_i^4(x)} < 0, \quad i = 1,\ldots,q. \end{aligned}$$

3. $s_n(x) < 0$

Similarly, in this case we can show that

$$a_i(x) + b_i(x)p(x) < 0, \quad i = 1,\ldots,q.$$

These discussions imply that Eq. (9) holds. That is, all the possible closed-loop systems satisfy the L_2-gain requirement specified in Eq. (5).

B. Internal stability

To prove internal stability, notice that Eq. (6) implies that, along the trajectories of system S_i under $w = 0$,

$$\begin{aligned} H_i(x,0,p(x)) &= \frac{\partial V(x)}{\partial x}\left(f_i(x) + g_{2i}(x)p(x)\right) + h_{1i}(x)^T h_{1i}(x) \\ &\leq a_i(x) + b_i(x)p(x) < 0 \quad \forall x \neq 0. \end{aligned}$$

That is, for each $i \in \{1,\ldots,q\}$, along the trajectories of system S_i, we have

$$\dot{V}(x) = \frac{\partial V(x)}{\partial x}\left(f_i(x) + g_{2i}(x)p(x)\right) < 0 \quad \forall x \neq 0.$$

This shows that all the closed-loop systems are internally stable.

***Remark 1*:** The systems considered in reference [8] are special cases of the systems considered in this chapter. If we let $\theta_{ij}(\bar{x}_j) = 0, i = 1, 2, ..., q$, and j=1,2,…,n-1, the systems in Eq. (3) will reduce to the systems considered in reference [8]. On the other hand, in reference [17], it is assumed that $U(s)$ is in quadratic form. In this chapter, we relax this restrictive assumption.

***Remark 2*:** In this chapter, we consider the case that the controlled output z is independent of the control input u. In this situation, a much simpler formula (not a special case of the formula in reference [7]) is proposed for constructing simultaneous H^∞ controllers. In the case that the controlled output z depends on u, necessary and sufficient conditions for the existence of simultaneous H^∞ controllers and a formula for constructing simultaneous H^∞ controllers can be derived by the results in reference [7].

4. An illustrative example

Consider the following nonlinear systems:

$$S_i : \begin{cases} \dot{x}_1 = x_2 + \theta_{i1}(x) \\ \dot{x}_2 = \theta_{i2}(x) + \rho_i(x)w + \eta_i(x)u \\ z = h_{1i}(x) + k_{11i}(x)w, \end{cases} \qquad i = 1, 2, \text{and } 3 \tag{10}$$

where

$$\begin{aligned}
\theta_{11}(x) &= x_1, \theta_{21}(x) = x_1\sin(x_1),\ \theta_{31}(x) = -x_1\cos(x_1), \\
\theta_{12}(x) &= x_1^2 + x_2^3, \theta_{22}(x) = x_1(1-2x_2),\ \theta_{32}(x) = x_1\cos(x_2) + 2x_2\sin(5x_1), \\
\rho_1(x) &= -1 + x_1,\ \rho_2(x) = x_1x_2,\ \rho_3(x) = x_1 - x_2^2, \\
\eta_1(x) &= 1 + (x_1 + x_2)^2,\ \eta_2(x) = 2 - \cos(x_1),\ \eta_3(x) = 2 + x_2^2, \\
h_{11}(x) &= x_1\cos(x_2^2),\ h_{12}(x) = -x_1\sin(x_1),\ h_{13}(x) = x_2, \\
k_{111}(x) &= -1 + \cos(x_1),\ k_{112}(x) = 1,\ k_{113}(x) = 1 + \sin(5x_2).
\end{aligned}$$

It can be shown that

http://dx.doi.org/10.5772/64105

$$h_{1i}^T(x)h_{1i}(x) + h_{1i}^T(x)k_{11i}(x)\left(\gamma^2 - k_{11i}^T(x)k_{11i}(x)\right)^{-1} k_{11i}^T(x)h_{1i}(x) \le U(x), \quad i = 1,2, \text{ and } 3$$

with

$$U(x) = \frac{9}{5}x_1^2 + \frac{9}{5}x_2^2.$$

Let $\gamma = 3$. It can be verified that *Assumption 1* holds. Let

$$\begin{aligned} s_1(\bar{x}_1) &= x_1 \\ \varphi_1(x_1) &= -2x_1 \\ \mu_1(x_1) &= x_1^2 \\ s_2(\bar{x}_2) &= x_2 - \varphi_1(\bar{x}_1) = 2x_1 + x_2. \end{aligned}$$

Then,

$$\hat{V}(x) = \frac{1}{2}\left(s_1^2(x_1) + s_2^2(\bar{x}_2)\right)$$

is positive, definite, and radially unbounded. By choosing $K = 10$, it can be shown that

$$-K\mu_1(x_1) + U(x)\Big|_{s_2(\bar{x}_2)=0} < 0, \quad \forall x_1 \neq 0.$$

Therefore,

$$V(x) = K\hat{V}(x) = 5\left(s_1^2(x_1) + s_2^2(\bar{x}_2)\right)$$

is a common CSF for the three systems in Eq. (10). For $i = 1$, 2, and 3, define

$$\begin{aligned} a_i(x) =& -K\mu_1(x) + Ks_2(x)\left(s_1(x_1) + \theta_{i2}(x) - \frac{\partial\varphi_1(x_1)}{\partial x_1}(x_2 + \theta_{i1}(x))\right) + h_{1i}^T(x)h_{1i}(x) \\ &+ \left(\frac{K}{2}s_n(x)\rho_i^T(x) + k_{11i}^T(x)h_{1i}(x)\right)^T \left(\gamma^2 I - k_{11i}^T(x)k_{11i}(x)\right)^{-1}\left(\frac{K}{2}s_n(x)\rho_i^T(x) + k_{11i}^T(x)h_{1i}(x)\right) \\ b_i(x) =& K\eta_i(x)s_2(x) \end{aligned}$$

and (with $\beta_1 = \beta_2 = \beta_3 = 0.1$)

$$p_i(x) \equiv \begin{cases} -\dfrac{a_i(x) + \sqrt{a_i^2(x) + \beta_i b_i^4(x)}}{b_i(x)}, & \text{if } s_2(x) \neq 0 \\ 0, & \text{if } s_2(x) = 0. \end{cases}$$

From *Theorem 2*, the following controller

$$u = p(x) \equiv \begin{cases} \min\{p_1(x), p_2(x), p_3(x)\}, & \text{if } 2x_1 + x_2 > 0 \\ 0, & \text{if } 2x_1 + x_2 = 0 \\ \max\{p_1(x), p_2(x), p_3(x)\}, & \text{if } 2x_1 + x_2 < 0 \end{cases} \tag{11}$$

is a simultaneous H^∞ controller for the three systems in Eq. (10). With arbitrarily chosen disturbance inputs, **Figures 1–3** show the states, control inputs, disturbance inputs, and controlled outputs of these three systems starting at different initial states with the same controller defined in Eq. (11). It can be seen that all the three closed-loop systems are internally stable and satisfy the required L_2-gain specification. That is, the controller defined in Eq. (11) is indeed a simultaneous H^∞ controller for the three systems in Eq. (10).

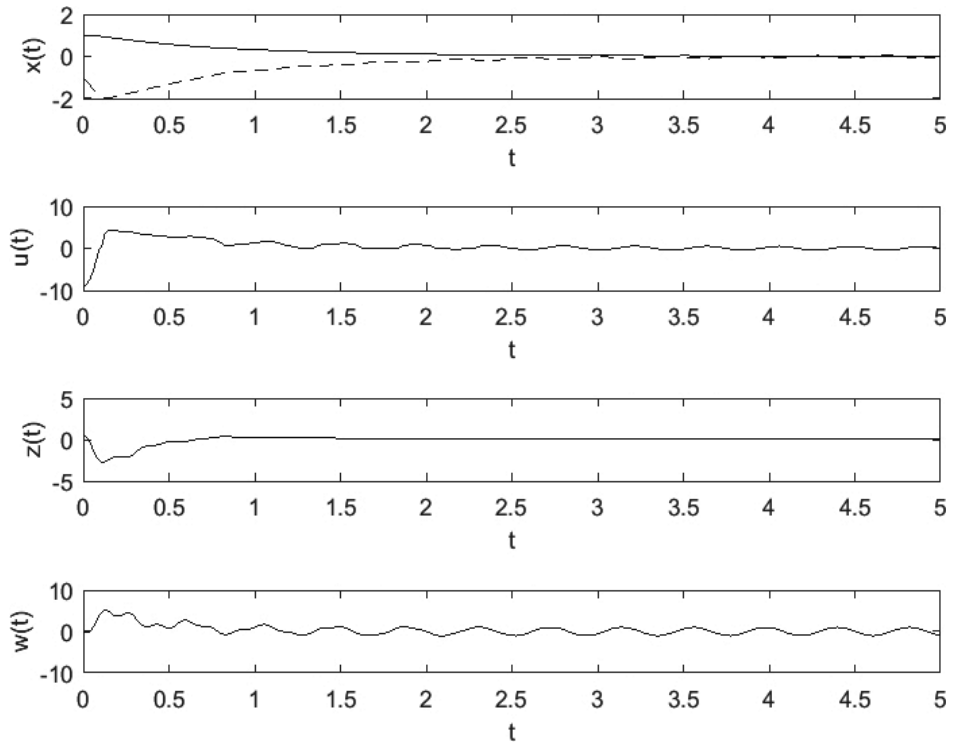

Figure 1. Responses of the system S_1 controlled by the controller defined in Eq. (11).

http://dx.doi.org/10.5772/64105

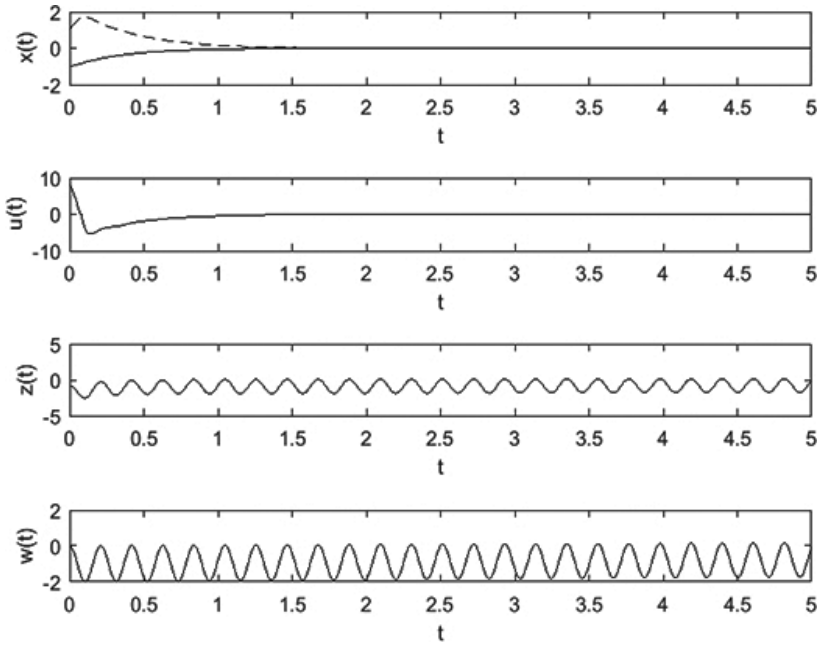

Figure 2. Responses of the system S_2 controlled by the controller defined in Eq. (11).

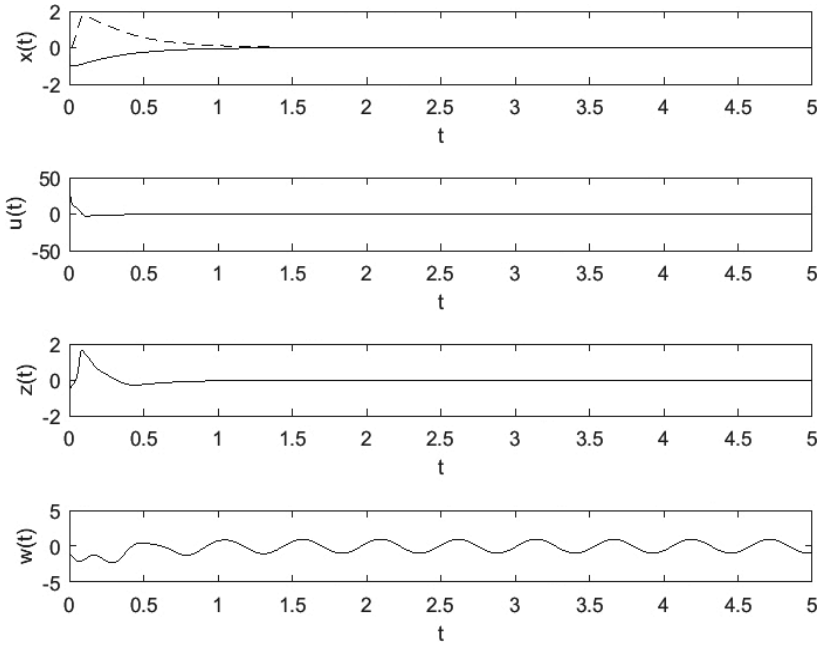

Figure 3. Responses of the system S_3 controlled by the controller defined in Eq. (11).

5. Conclusions

In this chapter, a systematic way for constructing simultaneous H^∞ state feedback controllers of nonlinear control systems in strict-feedback form is proposed. It is shown that the existence of common CSFs guarantees the existence of simultaneous H^∞ controllers. An explicit formula for constructing simultaneous H^∞ controllers is derived. The simulation example is given for verifying the theoretical results. The simulation results show, as expected, that the designed controller can simultaneously stabilize the considered systems and such that all closed-loop systems satisfy the specified disturbance attenuation requirement. Possible further works include considering nonlinear control systems in more general forms, applying the approach to time-varying case, and considering the output feedback case.

Author details

Jenq-Lang Wu[1*], Chee-Fai Yung[1] and Tsu-Tian Lee[2]

*Address all correspondence to: wujl@mail.ntou.edu.tw

1 Department of Electrical Engineering, National Taiwan Ocean University, Keelung, Taiwan, Republic of China

2 Department of Electrical Engineering, Tamkang University, New Taipei City, Taiwan, Republic of China

References

[1] Savkin, A. V. (1998). Simultaneous H^∞ control of a finite collection of linear plants with a single nonlinear digital controller. Systems & Control Letters. 33:281-289.

[2] Savkin, A. V. (1999). The problem of simultaneous H^∞ control. Applied Mathematics Letters. 12(1):53-56.

[3] Sebe, N. (1999). A design of controllers for simultaneous H^∞ control problem. International Journal of Systems Science. 30:25-31.

[4] Cao, Y. Y. and Lam, J. (2000). On simultaneous H^∞ control and strong H^∞ stabilization. Automatica. 36:859-865.

[5] Miller, D. E. and Chen, T. (2002). Simultaneous stabilization with near-optimal H^∞ performance. IEEE Transactions on Automatic Control. 47:1986-1998.

[6] Lee, P. H. and Soh, Y. C. (2004). Simultaneous H^∞ stabilization. International Journal of Control. 77:111-117.

http://dx.doi.org/10.5772/64105

[7] Wu, J. L. (2009). Simultaneous H^∞ control for nonlinear systems. IEEE Transactions on Automatic Control. 54(3):606-610.

[8] Wu, J. L. and Yung, C. F. (2010). Simultaneous H^∞ control for a collection of nonlinear systems in canonical form. The 19th Chinese Control Conference. Beijing, China. July 2010.

[9] Yung, C. F. and Wu, J. L. (2006). Control storage function approach to nonlinear L_2-gain control problem. The 6th Asian Control Conference, 71-77. Bali, Indonesia. July 2006.

[10] Artstein, Z. (1983). Stabilization with relaxed control. Nonlinear Analysis, Theory, Methods & Applications. 7:1163-1173.

[11] Battilotti, S. (1999). Robust stabilization of nonlinear systems with pointwise norm-bounded uncertainties: A control Lyapunov function approach. IEEE Transactions on Automatic Control. 44:3-17.

[12] Kokotovic, P. P. and Arcak, M. (2001). Constructive nonlinear control: a historical perspective. Automatica. 37:637-662.

[13] Krstic, M., Kanellakopoulos, I., & Kokotovic, P. (1995). Nonlinear and Adaptive Control Design. John Wiley & Sons, Inc. New York.

[14] Malisoff, M. and Sontag, E. D. (2000). Universal formulas for feedback stabilization with respect to Minkowski balls. Systems & Control Letters. 40:247-260.

[15] Sontag, E. D. (1983). A Lyapunov-like characterization of asymptotic controllability. SIAM Journal of Control and Optimization. 21:462-471.

[16] Sontag, E. D. (1989). A 'universal' constructive of Artstein's theorem on nonlinear stabilization. Systems & Control Letters. 12:542-550.

[17] Tsinias, J. (1990). Asymptotic feedback stabilization: A sufficient condition for the existence of control Lyapunov functions. Systems & Control Letters. 15:441-448.

[18] Wu, J. L. (2005). Simultaneous stabilization for a collection of single input nonlinear systems. IEEE Transactions on Automatic Control. 50(3):328-337.

Chapter 10

Nonlinear Feedback Control of Underactuated Mechanical Systems

Le Anh Tuan and Soon-Geul Lee

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64739

Abstract

This chapter presents control of a class of mechanical underactuated system using feedback linearization technique. The MIMO mechanical system is modeled by a set of nonlinear differential equations in which mathematical model is divided into two subsystems: one for actuated outputs and the other for unactuated outputs. The nonlinear feedback of states is used to "linearize" the closed-loop system. In other word, the control structure is constructed by linearly combining two components that are separately obtained from the nonlinear feedback of actuated and unactuated states. Lyapunov technique will be applied to investigate the system stability. As illustration example, nonlinear feedback control of a three-dimensional (3D) overhead crane is presented to investigate the proposed theory.

Keywords: underactuated mechanical systems, feedback linearization, Lyapunov's linearization theorem, overhead cranes

1. Introduction

In practice, many control problems involve the "underactuated" behavior of mechanical systems. In underactuated systems, the number of equipped actuators is less than that of the controlled variables. That is, actuators do not directly control several degrees of freedom. For example, we consider a tracking control problem for a marine vessel (**Figure 1**). In many cases, ships are equipped with either two independent aft thrusters or one main aft thruster and one rudder, without any bow or side thruster. Therefore, no sway control force acting on the ship is assumed. From the aforementioned condition, Lefeber et al. [1] investigated tracking control for underactuated ships in which three state variables, namely, surge, sway, and yaw, are

driven by only two inputs: surge force and yaw torque. We can find many underactuated systems in engineering, such as mobile robots, aircraft, and gantry cranes, among others.

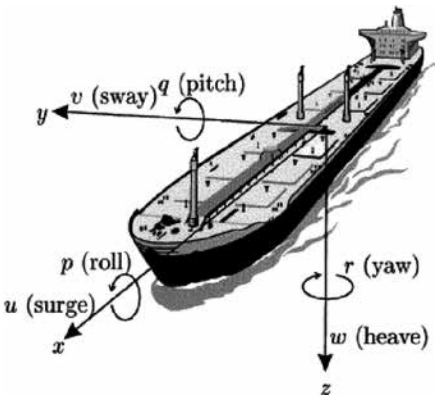

Figure 1. Tracking control of an underactuated ship [1].

According to the study of Tedrake [2], a mechanical system that can be described mathematically by

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}+\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\mathbf{G}(\mathbf{q})=\mathbf{B}(\mathbf{q})\mathbf{u} \tag{1}$$

is regarded an underactuated system if the rank of matrix $\mathbf{B}(\mathbf{q})$ is less than the dimension of vector $\mathbf{q}$, that is,

$$\operatorname{rank}\left(\mathbf{B}(\mathbf{q})\right)<\dim(\mathbf{q}). \tag{2}$$

Otherwise, system (1) has a "fully actuated" property in configuration $(\mathbf{q},\dot{\mathbf{q}},t)$ if it can control instantaneous acceleration in an arbitrary direction in $\mathbf{q}$.

$$\operatorname{rank}\left(\mathbf{B}(\mathbf{q})\right)=\dim(\mathbf{q}) \tag{3}$$

Unlike modern control techniques, such as fuzzy logic and neural networks, traditional control methods require knowing the physical properties of a system, which are generally governed by its mathematical model. For dynamical systems, a mathematical model is constructed based on mechanics principles, such as Newton's law, Lagrange equation, Lagrange multiplier method, Euler-Lagrange methodology, and so on. In mechanical systems with multiple degrees of freedom, system dynamics will comprise a set of second-order differential equations (1) in terms of displacements $\mathbf{q}$, velocities $\dot{\mathbf{q}}$, and time t. From this point of view, dynamical systems can be classified according to the type of mathematical model.

http://dx.doi.org/10.5772/64739

Partial differential equations are used to describe *distributed systems* mathematically, whereas *ordinary differential equations* govern the motions of *discrete systems*.

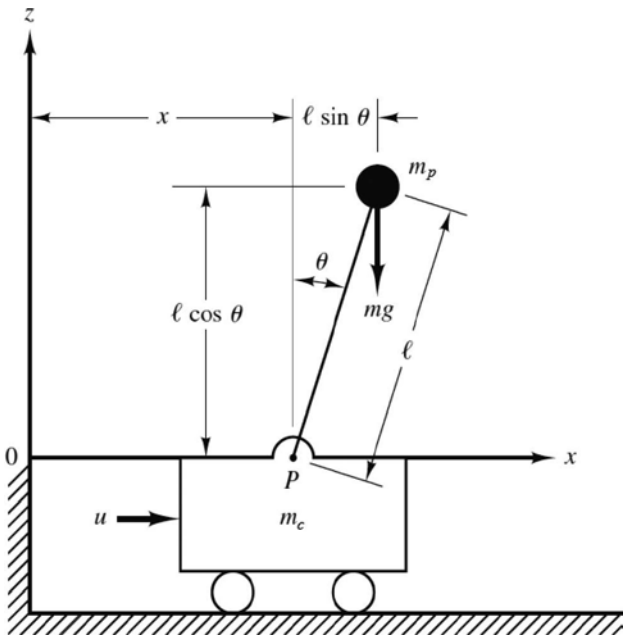

Figure 2. Cart-pole system [3].

Most realistic systems exhibit nonlinear behavior. A nonlinear system is generally described by nonlinear differential equations. Nonlinearities appear in a mathematical model because of its nonlinear components or geometric relationship. For example, a system that consists of an inverted pendulum mounted on a cart, as shown in **Figure 2**, has the following equations of motion:

$$\left(m_c + m_p\right)\ddot{x} + m_p l\cos\theta\ddot{\theta} - m_c l\dot{\theta}^2\sin\theta = u\,, \tag{4}$$

$$\cos\theta\ddot{x} + l\ddot{\theta} - g\sin\theta = 0\,. \tag{5}$$

The nonlinearities of the aforementioned dynamics originate from geometric constraint.

$$f\left(x, l, \sin\theta, \cos\theta\right) = 0 \tag{6}$$

The other example is a spring-damper system, which is illustrated in **Figure 3**. The force of nonlinear spring

$$F = k_1 x + k_2 x^3 \tag{7}$$

leads to the nonlinear modeling of the system, as follows:

$$m\ddot{x} + b\dot{x} + k_1 x + k_2 x^3 = u\ . \tag{8}$$

The nonlinear feedback technique, also called feedback linearization, is a representative method for controlling nonlinear systems. The main concept of feedback linearization is to transfer the original nonlinear system algebraically into the linear system by inserting equivalent inputs to suppress the nonlinearities of the former. The feedback linearization control of fully actuated systems has been discussed in several well-known textbooks [4, 5] in which this theory has been completely developed. Previous studies have pointed out that fully actuated systems are feedback linearizable through nonlinear feedback [6, 7]. In this chapter, we introduce the feedback linearization control for a class of multiple-input and multiple-output (MIMO) underactuated systems. The analysis process is conducted using an algebra foundation in which the mathematical model is simplified through matrix equations.

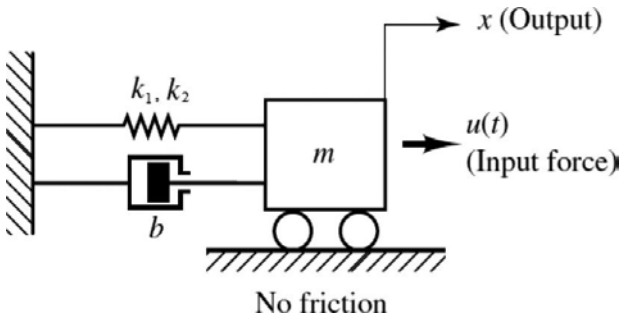

Figure 3. Mechanical system with a viscous damper and a nonlinear spring [3].

First, the mathematical model of underactuated mechanical systems is separated into two subsystems: actuated states and unactuated states. Then, we design a controller in which nonlinear feedback is partly applied to both actuated and unactuated dynamics. Subsequently, actuated submodel is "linearized" using a nonlinear feedback method; thus, the unactuated dynamics is regarded as internal model. Seeing actuated states as system outputs, a nonlinear control law is designed to drive state trajectories to the references. However, this controller does not promise the stability of unactuated states. Therefore, its structure should be adjusted to guarantee the stability of both actuated and unactuated states based on the nonlinear feedback of all system states. The control scheme now exhibits the linear combination of two components that are distinctly acquired from the nonlinear feedback of both the actuated and unactuated submodels.

In comparison with traditional controllers, such as the proportional-integral-derivative (PID) controller, partial feedback linearization (PFL) exhibits several advantages. In the PID con-

troller design, most of the nonlinear factors of a system are not mentioned. By contrast, in the design of PFL, all the nonlinearities of a system considered in the system dynamics are entirely vanished by the PFL controller. However, the PFL approach requires a precise model to achieve good control action. Additionally, the approach is not convenient in systems with uncertain parameters.

As an enhancement of Tuan et al.'s [8] paper, where PFL was applied for three-dimensional (3D) overhead crane, we introduce the PFL theory in the generalized form for a class of nonlinear underactuated mechanical systems. The outline of this chapter is as follows. Section 1 introduces the chapter. Section 2 presents the general form of the mathematical modeling of an underactuated mechanical system. Section 3 constructs a nonlinear controller based on the partial nonlinear feedback technique. Section 4 discusses system stability. Section 5 provides an example to illustrate the proposed theory. Finally, Section 6 provides the conclusion of the chapter.

2. Mathematical model

In general, the physical behavior of a MIMO mechanical system is governed by a set of differential equations of motion. Consider an underactuated system with n degrees of freedom driven by m actuators ($m<n$). The mathematical model, which is composed of n ordinary differential equations, is simplified in matrix form as follows:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{F}, \tag{9}$$

where $\mathbf{q} = \begin{bmatrix} q_1 & q_2 & \cdots & q_n \end{bmatrix}^T \in R^n$ is the vector of the generalized coordinates, and $\mathbf{F} \in R^n$ denotes the vector of the control inputs. Given that the system has more control signals than actuators, $\mathbf{F}$ has only m nonzero components as $\mathbf{F} = \begin{bmatrix} \mathbf{U} & \mathbf{0}_{(n-m)\times 1} \end{bmatrix}^T$, with $\mathbf{U} = \begin{bmatrix} u_1 & u_2 & \cdots & u_m \end{bmatrix}^T \in R^m$ being a vector of nonzero input forces. $\mathbf{M}(\mathbf{q}) = \mathbf{M}^T(\mathbf{q}) = [m_{ij}]_{n\times n} \in R^{n\times n}$ is the symmetric mass matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \left[c_{ij}\right]_{n\times n} \in R^{n\times n}$ is the Coriolis and centrifugal matrix, and $\mathbf{G}(\mathbf{q}) = \begin{bmatrix} g_1 & g_2 & \cdots & g_n \end{bmatrix}^T \in R^n$ indicates the gravity vector.

As an underactuated system, its n output signals are driven by m actuators. Meanwhile, its mathematical model is divided into two auxiliary dynamics, namely, actuated and unactuated systems. Correspondingly, $\mathbf{q}_a = \begin{bmatrix} q_1 & q_2 & \cdots & q_m \end{bmatrix}^T \in R^m$ for actuated states and $\mathbf{q}_u = \begin{bmatrix} q_{m+1} & \cdots & q_n \end{bmatrix}^T \in R^{n-m}$ for unactuated states are defined. The matrix differential equation (9) can then be divided into two equations as follows:

$$\mathbf{M}_{11}(\mathbf{q})\ddot{\mathbf{q}}_a + \mathbf{M}_{12}(\mathbf{q})\ddot{\mathbf{q}}_u + \mathbf{C}_{11}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a + \mathbf{C}_{12}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u + \mathbf{G}_1(\mathbf{q}) = \mathbf{U}, \tag{10}$$

$$\mathbf{M}_{21}(\mathbf{q})\ddot{\mathbf{q}}_a + \mathbf{M}_{22}(\mathbf{q})\ddot{\mathbf{q}}_u + \mathbf{C}_{21}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a + \mathbf{C}_{22}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u + \mathbf{G}_2(\mathbf{q}) = \mathbf{0}, \tag{11}$$

where $\mathbf{M}_{11}(\mathbf{q})$, $\mathbf{M}_{12}(\mathbf{q})$, $\mathbf{M}_{21}(\mathbf{q})$, $\mathbf{M}_{22}(\mathbf{q})$ are the submatrices of $\mathbf{M}(\mathbf{q})$; and $\mathbf{C}_{11}(\mathbf{q},\dot{\mathbf{q}})$, $\mathbf{C}_{12}(\mathbf{q},\dot{\mathbf{q}})$, $\mathbf{C}_{21}(\mathbf{q},\dot{\mathbf{q}})$, $\mathbf{C}_{22}(\mathbf{q},\dot{\mathbf{q}})$ are the submatrices of $\mathbf{C}_{11}(\mathbf{q},\dot{\mathbf{q}})$. Therefore, matrices $\mathbf{M}(\mathbf{q})$, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, and $\mathbf{G}(\mathbf{q})$ of Equation (9) exhibit the following form:

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} \mathbf{M}_{11}(\mathbf{q}) & \mathbf{M}_{12}(\mathbf{q}) \\ \mathbf{M}_{21}(\mathbf{q}) & \mathbf{M}_{22}(\mathbf{q}) \end{bmatrix}, \mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} \mathbf{C}_{11}(\mathbf{q},\dot{\mathbf{q}}) & \mathbf{C}_{12}(\mathbf{q},\dot{\mathbf{q}}) \\ \mathbf{C}_{21}(\mathbf{q},\dot{\mathbf{q}}) & \mathbf{C}_{22}(\mathbf{q},\dot{\mathbf{q}}) \end{bmatrix}, \mathbf{G}(\mathbf{q}) = \begin{bmatrix} \mathbf{G}_1(\mathbf{q}) \\ \mathbf{G}_2(\mathbf{q}) \end{bmatrix}.$$

Notably, matrix $\mathbf{M}(\mathbf{q})$ is symmetric positive definite, $\mathbf{M}_{12}(\mathbf{q}) = \mathbf{M}_{21}^T(\mathbf{q})$. The actuated equation (10) shows direct relationship between the actuated states $\mathbf{q}_a$ and the actuators $\mathbf{U}$. By contrast, the unactuated equation (11) does not display the constraint between the unactuated states $\mathbf{q}_u$ and the inputs $\mathbf{U}$. Physically, input signals $\mathbf{U}$ drive the actuated states $\mathbf{q}_a$ directly and the unactuated states $\mathbf{q}_u$ indirectly.

3. Nonlinear feedback control

System dynamics, which is composed of Equations (10) and (11), is transformed into a simpler model with an equivalent linear form based on the nonlinear feedback method [7]. Note that $\mathbf{M}_{22}(\mathbf{q})$ is a positive definite matrix. The unactuated states $\mathbf{q}_u$ can be determined from Equation (11) as

$$\ddot{\mathbf{q}}_u = -\mathbf{M}_{22}^{-1}(\mathbf{q})\left\{\mathbf{M}_{21}(\mathbf{q})\ddot{\mathbf{q}}_a + \mathbf{C}_{21}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a + \mathbf{C}_{22}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u + \mathbf{G}_2(\mathbf{q})\right\}. \tag{12}$$

In underactuated mechanical systems, the unactuated state $\mathbf{q}_u$ has a geometric relationship with the actuated state $\mathbf{q}_a$. Therefore, control input $\mathbf{U}$ indirectly acts on $\mathbf{q}_u$ through $\mathbf{q}_a$. Substituting Equation (12) into Equation (10) and simplifying the equation yield the following:

$$\bar{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}}_a + \bar{\mathbf{C}}_1(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a + \bar{\mathbf{C}}_2(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u + \bar{\mathbf{G}}_1(\mathbf{q}) = \mathbf{U}, \tag{13}$$

where

$\bar{\mathbf{M}}(\mathbf{q}) = \mathbf{M}_{11}(\mathbf{q}) - \mathbf{M}_{12}(\mathbf{q})\mathbf{M}_{22}^{-1}(\mathbf{q})\mathbf{M}_{21}(\mathbf{q})$,

$\bar{\mathbf{C}}_1(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}_{11}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{M}_{12}(\mathbf{q})\mathbf{M}_{22}^{-1}(\mathbf{q})\mathbf{C}_{21}(\mathbf{q},\dot{\mathbf{q}})$

$\bar{\mathbf{C}}_2(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}_{12}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{M}_{12}(\mathbf{q})\mathbf{M}_{22}^{-1}(\mathbf{q})\mathbf{C}_{22}(\mathbf{q},\dot{\mathbf{q}})$ and

$\bar{\mathbf{G}}_1(\mathbf{q}) = \mathbf{G}_1(\mathbf{q}) - \mathbf{M}_{12}(\mathbf{q})\mathbf{M}_{22}^{-1}(\mathbf{q})\mathbf{G}_2(\mathbf{q})$.

http://dx.doi.org/10.5772/64739

$\bar{\mathbf{M}}(\mathbf{q})$ is a positive definite matrix for every $\mathbf{q} = \begin{bmatrix}\mathbf{q}_a & \mathbf{q}_u\end{bmatrix}^T \in R^n$. Equation (13) is transformed into

$$\ddot{\mathbf{q}}_a = \bar{\mathbf{M}}^{-1}(\mathbf{q})\left\{\mathbf{U} - \bar{\mathbf{C}}_1(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a - \bar{\mathbf{C}}_2(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u - \bar{\mathbf{G}}_1(\mathbf{q})\right\}. \tag{14}$$

By inserting Equation (14) into Equation (12), we obtain

$$\ddot{\mathbf{q}}_u = -\mathbf{M}_{22}^{-1}(\mathbf{q})\left\{\mathbf{M}_{21}(\mathbf{q})\bar{\mathbf{M}}^{-1}(\mathbf{q})\mathbf{U} + \bar{\mathbf{C}}_3(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a + \bar{\mathbf{C}}_4(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u + \bar{\mathbf{G}}_2(\mathbf{q})\right\}, \tag{15}$$

where

$\bar{\mathbf{C}}_3(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}_{21}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{M}_{21}(\mathbf{q})\bar{\mathbf{M}}^{-1}(\mathbf{q})\bar{\mathbf{C}}_1(\mathbf{q},\dot{\mathbf{q}})$,

$\bar{\mathbf{C}}_4(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}_{22}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{M}_{21}(\mathbf{q})\bar{\mathbf{M}}^{-1}(\mathbf{q})\bar{\mathbf{C}}_2(\mathbf{q},\dot{\mathbf{q}})$,

and $\bar{\mathbf{G}}_2(\mathbf{q}) = \mathbf{G}_2(\mathbf{q}) - \mathbf{M}_{21}(\mathbf{q})\bar{\mathbf{M}}^{-1}(\mathbf{q})\bar{\mathbf{G}}_1(\mathbf{q})$.

Therefore, the dynamic behavior of a mechanical underactuated system can be described by actuated dynamics (14) and unactuated dynamics (15) in which the mathematical relationships among $\mathbf{q}_a$, $\mathbf{q}_u$, and $\mathbf{U}$ can be observed clearly.

Considering the actuated states $\mathbf{q}_a$ as the system outputs, actuated dynamics (14) can be "linearized" by defining

$$\ddot{\mathbf{q}}_a = \mathbf{V}_a, \tag{16}$$

with $\mathbf{V}_a \in R^m$ as the equivalent control inputs. Then, the control signals $\mathbf{U}$ become

$$\mathbf{U} = \bar{\mathbf{M}}(\mathbf{q})\mathbf{V}_a + \bar{\mathbf{C}}_1(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_a + \bar{\mathbf{C}}_2(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_u + \bar{\mathbf{G}}_1(\mathbf{q}). \tag{17}$$

Controller $\mathbf{U}$ is designed to drive the actuated states $\mathbf{q}_a$ to the desired values $\mathbf{q}_{ad}$. To track the given state trajectories, the following equivalent control inputs are selected:

$$\mathbf{V}_a = \ddot{\mathbf{q}}_{ad} - \mathbf{K}_{ad}(\dot{\mathbf{q}}_a - \dot{\mathbf{q}}_{ad}) - \mathbf{K}_{ap}(\mathbf{q}_a - \mathbf{q}_{ad}). \tag{18}$$

Given that $\mathbf{q}_{ad}$ = **const**, Equation (18) can be reduced into

$$\mathbf{V}_a = -\mathbf{K}_{ad}\dot{\mathbf{q}}_a - \mathbf{K}_{ap}(\mathbf{q}_a - \mathbf{q}_{ad}), \tag{19}$$

with $\mathbf{K}_{ad} = \mathrm{diag}(K_{ad1}, K_{ad2}, \ldots, K_{adm}) \in R^{m\times m}$, $\mathbf{K}_{ap} = \mathrm{diag}(K_{ap1}, K_{ap2}, \ldots, K_{apm}) \in R^{m\times m}$ as positive diagonal matrices.

On the basis of Equation (18) and active dynamics (16), the differential equation of the tracking error is obtained as described by

$$\ddot{\tilde{\mathbf{q}}}_a + \mathbf{K}_{ad}\dot{\tilde{\mathbf{q}}}_a + \mathbf{K}_{ap}\tilde{\mathbf{q}}_a = \mathbf{0}, \tag{20}$$

where $\tilde{\mathbf{q}}_a = \mathbf{q}_a - \mathbf{q}_{ad}$ is the tracking error vector of the actuated states. Evidently, the dynamics of the tracking error (20) is exponentially stable for every $\mathbf{K}_{ad} > \mathbf{0}$ and $\mathbf{K}_{ap} > \mathbf{0}$. That is, the tracking errors of the actuated states $\tilde{\mathbf{q}}_a$ approach zero (or $\mathbf{q}_a$ converges to $\mathbf{q}_{ad}$) as t becomes infinite. In particular, the equivalent control $\mathbf{V}_a$ forces the actuated states $\mathbf{q}_a$ to reach the references $\mathbf{q}_{ad}$ asymptotically.

The control scheme (17), which corresponds to the equivalent input $\mathbf{V}_a$, is used only to stabilize the actuated states $\mathbf{q}_a$ asymptotically. To stabilize the unactuated states $\mathbf{q}_u$, the nonlinear feedback technique can be applied to subdynamics (15) as follows:

$$\ddot{\mathbf{q}}_u = \mathbf{V}_u = -\mathbf{K}_{ud}\dot{\mathbf{q}}_u - \mathbf{K}_{up}\mathbf{q}_u, \tag{21}$$

where $\mathbf{V}_u \in R^{n-m}$ refers to the equivalent inputs of the unactuated states.

$\mathbf{K}_{ud}$ = diag (K_{ud1}, K_{ud2}, …, $K_{ud(n-m)}$) $\in R^{(n-m)\times(n-m)}$ and $\mathbf{K}_{up}$ = diag (K_{up1}, K_{up2}, …, $K_{up(n-m)}$) $\in R^{(n-m)\times(n-m)}$ are positive matrices.

The control input $\mathbf{U}$ received from Equations (15) and (21) ensures the stability of the unactuated states $\mathbf{q}_u$ because the tracking error dynamics, that is,

$$\ddot{\mathbf{q}}_u + \mathbf{K}_{ud}\dot{\mathbf{q}}_u + \mathbf{K}_{up}\mathbf{q}_u = \mathbf{0}, \tag{22}$$

is stable for every $\mathbf{K}_{ud} > \mathbf{0}$ and $\mathbf{K}_{up} > \mathbf{0}$. Hence, if $\mathbf{K}_{ud}$ and $\mathbf{K}_{up}$ are selected appropriately, then the equivalent inputs $\mathbf{V}_u$ can drive cargo swings $\mathbf{q}_u$ toward zero.

To stabilize the unactuated and actuated states, overall equivalent inputs are proposed by linearly combining $\mathbf{V}_a$ and $\mathbf{V}_u$ as follows:

$$\begin{aligned}\mathbf{V} &= \mathbf{V}_a + \boldsymbol{\alpha}\mathbf{V}_u \\ &= -\mathbf{K}_{ad}\dot{\mathbf{q}}_a - \mathbf{K}_{ap}\left(\mathbf{q}_a - \mathbf{q}_{ad}\right) - \boldsymbol{\alpha}\left(\mathbf{K}_{ud}\dot{\mathbf{q}}_u + \mathbf{K}_{up}\mathbf{q}_u\right)'\end{aligned} \tag{23}$$

with $\boldsymbol{\alpha} \in R^{m\times(n-m)}$ being the weighting matrix and $\mathbf{V} \in R^m$.

Hence, considering $\mathbf{q}_a$ as the primary output, the total control scheme is determined by replacing $\mathbf{V}_a$ with $\mathbf{V}$ in Equation (17). By substituting Equation (23) into Equation (17), the nonlinear feedback control structure is obtained as

http://dx.doi.org/10.5772/64739

$$\begin{aligned}\mathbf{U} = &\left(\bar{\mathbf{C}}_1\left(\mathbf{q},\dot{\mathbf{q}}\right)-\bar{\mathbf{M}}\left(\mathbf{q}\right)\mathbf{K}_{ad}\right)\dot{\mathbf{q}}_a+\left(\bar{\mathbf{C}}_2\left(\mathbf{q},\dot{\mathbf{q}}\right)-\bar{\mathbf{M}}\left(\mathbf{q}\right)\alpha\mathbf{K}_{ud}\right)\dot{\mathbf{q}}_u \\ &-\bar{\mathbf{M}}\left(\mathbf{q}\right)\mathbf{K}_{ap}\left(\mathbf{q}_a-\mathbf{q}_{ad}\right)-\bar{\mathbf{M}}\left(\mathbf{q}\right)\alpha\mathbf{K}_{up}\mathbf{q}_u+\bar{\mathbf{G}}_1\left(\mathbf{q}\right)\end{aligned}. \tag{24}$$

The nonlinear controller (23) asymptotically stabilizes all system state trajectories, as illustrated in an example presented in Section 5.

4. Analysis of system stability

The control law **U** is obtained from the actuated dynamics (14). The stability of the remaining part (the unactuated dynamics) of the closed-loop system, called the internal dynamics, is analyzed. If the internal dynamics is stable, then the tracking control problem is solved. Substituting the control scheme (24) into the unactuated subsystem (15) yields the internal dynamics:

$$\ddot{\mathbf{q}}_u = -\mathbf{M}_{22}^{-1}\left(\mathbf{q}\right)\begin{Bmatrix}\left(\mathbf{C}_{21}\left(\mathbf{q},\dot{\mathbf{q}}\right)-\mathbf{M}_{21}\left(\mathbf{q}\right)\mathbf{K}_{ad}\right)\dot{\mathbf{q}}_a+\left(\mathbf{C}_{22}\left(\mathbf{q},\dot{\mathbf{q}}\right)-\mathbf{M}_{21}\left(\mathbf{q}\right)\alpha\mathbf{K}_{ud}\right)\dot{\mathbf{q}}_u \\ -\mathbf{M}_{21}\left(\mathbf{q}\right)\mathbf{K}_{ap}\left(\mathbf{q}_a-\mathbf{q}_{ad}\right)-\mathbf{M}_{21}\left(\mathbf{q}\right)\alpha\mathbf{K}_{up}\mathbf{q}_u+\mathbf{G}_2\left(\mathbf{q}\right)\end{Bmatrix} \tag{25}$$

The local stability of the internal dynamics is guaranteed if the zero dynamics is exponentially stable. Setting $\mathbf{q}_a = \mathbf{q}_{ad}$ in the internal dynamics (25), the zero dynamics of the system is obtained as

$$\ddot{\mathbf{q}}_u+\mathbf{M}_{22}^{-1}\left(\mathbf{q}\right)\left\{\left(\mathbf{C}_{22}\left(\mathbf{q}_u,\dot{\mathbf{q}}_u\right)-\mathbf{M}_{21}\left(\mathbf{q}_u\right)\alpha\mathbf{K}_{ud}\right)\dot{\mathbf{q}}_u-\mathbf{M}_{21}\left(\mathbf{q}_u\right)\alpha\mathbf{K}_{up}\mathbf{q}_u+\mathbf{G}_2\left(\mathbf{q}_u\right)\right\}=\mathbf{0}. \tag{26}$$

The zero dynamics is expanded into a set of (n–m) second-order nonlinear differential equations in which the (n–m) components of vector $\mathbf{q}_u$ are considered as variables. The stability of the zero dynamics (26) is analyzed using Lyapunov's linearization theorem [4]. By defining 2(n–m) state variables $\mathbf{z} \in R^{2\times(n-m)}$, the zero dynamics (26) is converted into state-space form as follows:

$$\dot{\mathbf{z}} = \mathbf{f}\left(\mathbf{z}\right), \tag{27}$$

where $\mathbf{f}(\mathbf{z})$ is a vector of nonlinear functions, and $\mathbf{z} \in R^{2\times(n-m)}$ is a state vector. System dynamics (27) is composed of 2(n–m) first-order nonlinear differential equations. This nonlinear zero dynamics is asymptotically stable around the equilibrium point $\mathbf{z} = \mathbf{0}$ ($\mathbf{q}_u = \dot{\mathbf{q}}_u = \mathbf{0}$) if the corresponding linearized system is strictly stable. Linearizing the zero dynamics around $\mathbf{z} = \mathbf{0}$ yields a linearized system in the following form:

$$\dot{\mathbf{z}} = \mathbf{A}\mathbf{z}\,, \tag{28}$$

with

$$\mathbf{A} = \left(\frac{\partial \mathbf{f}}{\partial \mathbf{z}}\right)_{\mathbf{z}=\mathbf{0}} \tag{29}$$

as a 2(n–m)×2(n–m) Jacobian matrix of components $\partial f_i/\partial x_j$. The stability of the linear system (28) can be analyzed by considering the positions of the eigenvalues of **A** or using several traditional techniques, such as the Routh-Hurwitz criterion [3], the root locus method, and so on. Thus, by investigating the stability of the linear system (28), we can understand the dynamic behavior of the nonlinear system (27), or equivalently, zero dynamics (26), according to Lyapunov's linearization theorem [4].

The nonlinear system (26) *is asymptotically stable around the equilibrium point* ($\mathbf{q}_u = \dot{\mathbf{q}}_u = \mathbf{0}$) *if the linearized system* (28) *is strictly stable*.

The equilibrium point ($\mathbf{q}_u = \dot{\mathbf{q}}_u = \mathbf{0}$) *of the nonlinear system* (26) *is unstable if the linearized system* (28) *is unstable*.

We cannot conclude the stability of the nonlinear system (26) *if the linearized system* (28) *is marginally stable*.

As we will see in the examples provided in Section 5, the analysis of system stability using the aforementioned theorem yields the constraint equations of the controller parameters.

5. An application example

We apply the aforementioned theory to a 3D crane system to understand the proposed methodology comprehensively.

5.1. Problem statement

An overhead crane is a symbol of underactuated mechanical systems. Overhead cranes are typically used to transport cargo over short distances or to small areas, such as automotive factories and shipyards. We have investigated the nonlinear feedback control problem for a 3D overhead crane [8] with three actuators used to stabilize five outputs. The crane system, which is composed of four masses, is physically modeled in **Figure 4**. The distributed masses of the bridge are converted into a concentrated mass m_b, which is placed at the center of the bridge. m_l denotes the equivalent mass of the hoist mechanism, whereas m_t and m_c are the masses of the trolley and cargo, respectively. The system includes five degrees of freedom, which correspond to five generalized coordinates. $x(t)$ is the trolley motion, $z(t)$ is the bridge movement, and cargo position is characterized by three generalized coordinates (l, θ, and ϕ). Therefore, the generalized coordinates of the system are described by $\mathbf{q} = [z\ x\ l\ \varphi\ \theta]^T$.

http://dx.doi.org/10.5772/64739

Additionally, the friction of cargo hoisting, as well as trolley and bridge motions, is linearly characterized by damping factors b_r, b_t, and b_b, respectively. The control signals u_b, u_t and u_l correspondingly demonstrate the driving forces of trolley motion, bridge movement, and cargo lifting translation.

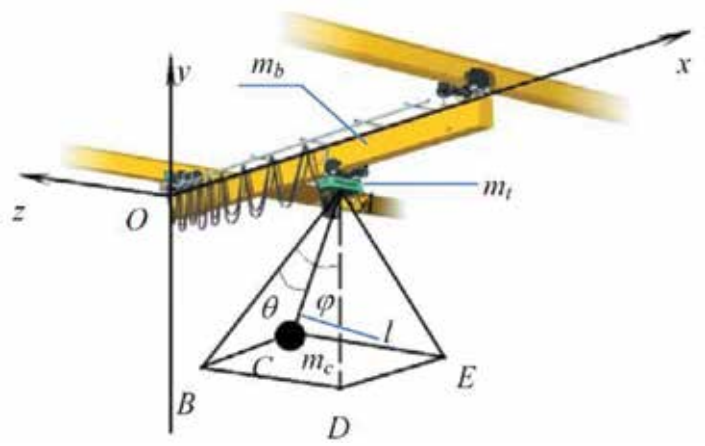

Figure 4. Physical modeling of a 3D overhead crane.

The main objective of this example is to design a controller for simultaneously conducting five tasks: (1) tracking the bridge, (2) moving the trolley to its destinations, (3) lifting/lowering the payload to the desired length of the cable, (4) keeping the cargo swing angles small during transportation, and (5) completely suppressing these swings at payload destinations.

By using Lagrange's equation to constitute the mathematical model, overhead crane dynamics can be represented by matrix equation (9) in which the component matrices are determined by the following formulas:

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} m_{11} & 0 & m_{13} & m_{14} & m_{15} \\ 0 & m_{22} & m_{23} & 0 & m_{25} \\ m_{31} & m_{32} & m_{33} & 0 & 0 \\ m_{41} & 0 & 0 & m_{44} & 0 \\ m_{51} & m_{52} & 0 & 0 & m_{55} \end{bmatrix}, \mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} b_b & 0 & c_{13} & c_{14} & c_{15} \\ 0 & b_t & c_{23} & 0 & c_{25} \\ 0 & 0 & b_r & c_{34} & c_{35} \\ 0 & 0 & c_{43} & c_{44} & c_{45} \\ 0 & 0 & c_{53} & c_{54} & c_{55} \end{bmatrix},$$

$$\mathbf{F} = \begin{bmatrix} u_b & u_t & u_l & 0 & 0 \end{bmatrix}^T, \mathbf{G}(\mathbf{q}) = \begin{bmatrix} 0 & 0 & g_3 & g_4 & g_5 \end{bmatrix}.$$

The coefficients of the $\mathbf{M}(\mathbf{q})$ matrix are given by

$$m_{11} = m_t + m_b + m_c,\ m_{13} = m_{31} = m_c \sin\varphi\cos\theta,\ m_{14} = m_{41} = m_c l\cos\varphi\cos\theta,$$

$m_{15} = m_{51} = -m_c l\sin\varphi\sin\theta,\ m_{22} = m_t + m_c,\ m_{23} = m_c\sin\theta,\ m_{25} = m_{52} = m_c l\cos\theta,$

and $m_{32} = m_c\sin\theta,\ m_{33} = m_l + m_c,\ m_{44} = m_c l^2\cos^2\theta,\ m_{55} = m_c l^2$.

The coefficients of the $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ matrix are determined by

$c_{13} = m_c\cos\varphi\cos\theta\dot{\varphi} - m_c\sin\varphi\sin\theta\dot{\theta},$

$c_{14} = m_c\cos\varphi\cos\theta\dot{l} - m_c l\cos\varphi\sin\theta\dot{\theta} - m_c l\sin\varphi\cos\theta\dot{\varphi},$

$c_{15} = -m_c l\cos\varphi\sin\theta\dot{\varphi} - m_c\sin\varphi\sin\theta\dot{l} - m_c l\sin\varphi\cos\theta\dot{\theta},$

$c_{23} = m_c\cos\theta\dot{\theta},\ c_{25} = m_c\cos\theta\dot{l} - m_c l\sin\theta\dot{\theta},\ c_{34} = -m_c l\cos^2\theta\dot{\varphi},$

$c_{35} = -m_c l\dot{\theta},\ c_{43} = m_c l\cos^2\theta\dot{\varphi},\ c_{44} = m_c l\cos^2\theta\dot{l} - m_c l^2\cos\theta\sin\theta\dot{\theta},$ and

$c_{45} = -m_c l^2\cos\theta\sin\theta\dot{\varphi},\ c_{53} = m_c l\dot{\theta},\ c_{54} = m_c l^2\cos\theta\sin\theta\dot{\varphi},\ c_{55} = m_c l\dot{l}$.

The nonzero coefficients of the $\mathbf{G}(\mathbf{q})$ vector are given by

$$g_3 = -m_c g\cos\varphi\cos\theta,\ g_4 = m_c gl\sin\varphi\cos\theta,\ g_5 = m_c gl\cos\varphi\sin\theta$$

5.2. Controller design

The overhead crane is an underactuated system in which five output signals are driven by three actuators. Using the nonlinear feedback methodology, we construct a control law

$$\mathbf{F} = \begin{bmatrix}\mathbf{U} & \mathbf{0}_{2\times1}\end{bmatrix}^T, \tag{30}$$

with $\mathbf{U} = \begin{bmatrix}u_b & u_t & u_l\end{bmatrix}^T$ to drive the actuated states $\mathbf{q}_a = [z\ x\ l]^T$ to the desired destinations $\mathbf{q}_{ad} = \begin{bmatrix}z_d & x_d & l_d\end{bmatrix}^T$ and the actuated states (cargo swings) $\mathbf{q}_2 = [\varphi\ \theta]^T$ toward zero.

Applying the theory proposed in Sections 1–4, we determine the structure of the controller in Equation (24), where $\mathbf{K}_{ad}$ = diag(K_{ad1}, K_{ad2}, K_{ad3}), $\mathbf{K}_{ap}$ = diag(K_{ap1}, K_{ap2}, K_{ap3}), $\mathbf{K}_{ud}$ = diag(K_{ud1}, K_{ud2}), and $\mathbf{K}_{up}$ = diag(K_{up1}, K_{up2}) are the positive matrices of control gains, and $\boldsymbol{\alpha} = \begin{bmatrix}\alpha_1 & 0\\ 0 & \alpha_2\\ 0 & 0\end{bmatrix}$ is a weighting matrix.

http://dx.doi.org/10.5772/64739

5.3. System stability

As presented in Section 4, we analyze the local stability of the internal dynamics (25), or equivalently, the zero dynamics (26). Applying Equation (26) to a 3D overhead crane, the zero dynamics of the system is expanded as

$$l_d\ddot{\varphi} - 2l_d \tan\theta\dot{\theta}\dot{\varphi} - \alpha_1 K_{ud1}\frac{\cos\varphi}{\cos\theta}\dot{\varphi} - \alpha_1 K_{up1}\frac{\cos\varphi}{\cos\theta}\varphi + g\frac{\sin\varphi}{\cos\theta} = 0\,, \tag{31}$$

$$\begin{pmatrix} l_d\ddot{\theta} + l_d\cos\theta\sin\theta\dot{\varphi}^2 + \alpha_1 K_{ud1}\sin\varphi\sin\theta\dot{\varphi} - \alpha_2 K_{ud2}\cos\theta\dot{\theta} \\ +\alpha_1 K_{up1}\sin\varphi\sin\theta\varphi - \alpha_2 K_{up2}\cos\theta\theta + g\cos\varphi\sin\theta \end{pmatrix} = 0\,. \tag{32}$$

The stability of the zero dynamics, which comprises Equations (31) and (32), is analyzed using Lyapunov's linearization theorem. First, we represent the zero dynamics in the first-order form by setting the four state variables as

$$z_1 = \varphi, z_2 = \dot{\varphi},\ z_3 = \theta, z_4 = \dot{\theta}$$

Then, the zero dynamics exhibits the following state-space forms:

$$\dot{z}_1 = z_2\,, \tag{33}$$

$$\dot{z}_2 = \left(2\tan z_3 z_4 z_2 + \frac{\alpha_1 K_{ud1}}{l_d}\frac{\cos z_1}{\cos z_3}z_2 + \frac{\alpha_1 K_{up1}}{l_d}\frac{\cos z_1}{\cos z_3}z_1 - \frac{g}{l_d}\frac{\sin z_1}{\cos z_3}\right) = f(\mathbf{z})\,, \tag{34}$$

$$\dot{z}_3 = z_4\,, \tag{35}$$

$$\dot{z}_4 = \begin{pmatrix} -\cos z_3\sin z_3 z_2^2 - \dfrac{\alpha_1 K_{ud1}}{l_d}\sin z_1\sin z_3 z_2 + \dfrac{\alpha_2 K_{ud2}}{l_d}\cos z_3 z_4 \\ -\dfrac{\alpha_1 K_{up1}}{l_d}\sin z_1\sin z_3 z_1 + \dfrac{\alpha_2 K_{up2}}{l_d}\cos z_3 z_3 - \dfrac{g}{l_d}\cos z_1\sin z_3 \end{pmatrix} = h(\mathbf{z})\,. \tag{36}$$

Using $\mathbf{z} = \begin{bmatrix} z_1 & z_2 & z_3 & z_4 \end{bmatrix}^T$ as the state vector, the nonlinear zero dynamics (33)–(36) are asymptotically stable around the equilibrium point $\mathbf{z} = \mathbf{0}$ ($\mathbf{q}_u = \dot{\mathbf{q}}_u = \mathbf{0}$) if the linearized system is strictly stable. Linearizing the zero dynamics around $\mathbf{z} = \mathbf{0}$ leads to a linear system as follows:

$$\dot{\mathbf{z}} = \mathbf{A}\mathbf{z}, \tag{37}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial f}{\partial z_1} & \frac{\partial f}{\partial z_2} & \frac{\partial f}{\partial z_3} & \frac{\partial f}{\partial z_4} \\ 0 & 0 & 0 & 1 \\ \frac{\partial h}{\partial z_1} & \frac{\partial h}{\partial z_2} & \frac{\partial h}{\partial z_3} & \frac{\partial h}{\partial z_4} \end{bmatrix}_{\mathbf{z}=\mathbf{0}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\alpha_1 K_{up1} - g}{l_d} & \frac{\alpha_1 K_{ud1}}{l_d} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{\alpha_2 K_{up2} - g}{l_d} & \frac{\alpha_2 K_{ud2}}{l_d} \end{bmatrix} \tag{38}$$

is a Jacobian matrix in which the characteristic polynomial exhibits the following form:

$$\begin{aligned} \left|\mathbf{A} - s\mathbf{I}_4\right| = s^4 &- \frac{\left(\alpha_1 K_{ud1} + \alpha_2 K_{ud2}\right)}{l_d} s^3 + \left(\frac{\alpha_1 \alpha_2 K_{ud1} K_{ud2}}{l_d^2} - \frac{\alpha_1 K_{up1} + \alpha_2 K_{up2} - 2g}{l_d} \right) s^2 \\ &+ \frac{\alpha_1 \alpha_2 \left(K_{ud1} K_{up2} + K_{up1} K_{ud2} \right) - g\left(\alpha_1 K_{ud1} + \alpha_2 K_{ud2}\right)}{l_d^2} s \\ &+ \frac{\alpha_1 \alpha_2 K_{up1} K_{up2} + g^2 - g\left(\alpha_1 K_{up1} + \alpha_2 K_{up2}\right)}{l_d^2}. \end{aligned} \tag{39}$$

The linearized system (37) is stable around the equilibrium point $\mathbf{z} = \mathbf{0}$ if $\mathbf{A}$ is a Hurwitz matrix. On the basis of the Hurwitz's criterion and the results of the calculations, the constraint condition of the controller parameters is determined as

$$\alpha_1 K_{ud1} + \alpha_2 K_{ud2} < 0, \tag{40}$$

$$\alpha_1 \alpha_2 K_{ud1} K_{ud2} > l_d \left(\alpha_1 K_{up1} + \alpha_2 K_{up2} - 2g \right), \tag{41}$$

$$\alpha_1 \alpha_2 \left(K_{ud1} K_{up2} + K_{up1} K_{ud2} \right) > g\left(\alpha_1 K_{ud1} + \alpha_2 K_{ud2} \right), \tag{42}$$

$$\alpha_1 \alpha_2 K_{up1} K_{up2} + g^2 > g\left(\alpha_1 K_{up1} + \alpha_2 K_{up2} \right). \tag{43}$$

Therefore, if Equations (40)–(43) among the control parameters are maintained, then the zero dynamics is stable around the equilibrium point $\mathbf{z} = \mathbf{0}$, which leads to the local stability of the internal dynamics (25).

http://dx.doi.org/10.5772/64739

5.4. Simulation and experiment

The overhead crane dynamics (9) driven by the control inputs (30) is numerically simulated in the case of a crane system that involves complicated operations. Accordingly, the trolley is forced to move from its initial position to the desired displacement at 0.4 m. The bridge is driven from its starting point to the desired location at 0.3 m, and the cargo is lifted with a cable length of 1–0.7 m of cable reference. These processes (lifting the cargo, moving the trolley, and driving the bridge) must be initiated simultaneously, with the cargo suspension cable initially perpendicular to the ground. The parameters used for the simulation are listed in **Table 1**.

System dynamics	Controller
$g = 9.81$ m/s^2, $m_c = 0.85$ kg, $m_t = 5$ kg	$\mathbf{K}_{ad} = \text{diag}(1.5, 1.5, 2.5)$, $\mathbf{K}_{ud} = \text{diag}(3, 3)$
$m_b = 7$ kg, $m_l = 2$ kg, $b_t = 20$ Nm/s	$\mathbf{K}_{ap} = \text{diag}(0.85, 0.87, 2)$, $\mathbf{K}_{up} = \text{diag}(0.5, 0.5)$
$b_b = 30$ Nm/s, $b_r = 50$ Nm/s	$\alpha_1 = \alpha_2 = -1$

Table 1. Crane system parameters.

Figure 5. Overhead crane system used for the experiments.

Additionally, an experimental study is conducted to verify the simulation results. **Figure 5** shows a laboratory crane system used for the experiment. In this system, three DC motors for the bridge motion, trolley movement, and cargo hoisting motion are used. Five incremental encoders are applied for measuring bridge and trolley motions, the movement of the cargo along the cable, and the two swing angles of the cargo.

Three-dimensional overhead crane is controlled by a target PC in which a control structure is built based on MATLAB/SIMULINK with an xPC target foundation. A host PC is linked to the target PC, and the crane system is connected to the target PC by two interface cards. The 6602

card sends PWM signals to the motor amplifiers and obtains feedback pulses from the encoders. The 6025E multifunction card is utilized for sending direction control signals to the motor amplifiers.

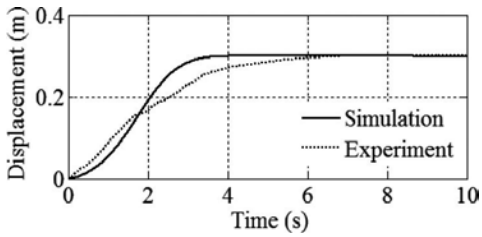

Figure 6. Bridge motion.

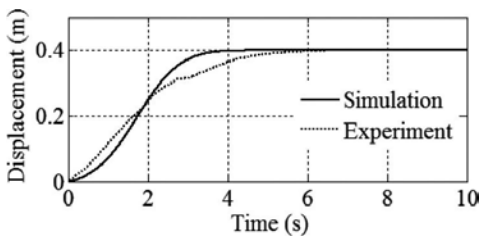

Figure 7. Trolley motion.

Figures 6–18 describe both the simulation and the experiment results. **Figures 6–8** show the paths of the bridge motion, trolley movement, and payload lifting translation, respectively. All the responses approach asymptotically to the destinations. However, the simulation paths are smoother and achieve steady states earlier than the experiment ones. The bridge moves and stops accurately at the load endpoint after 4 s in the simulation and 6 s in the experiment. The trolley reaches its destination after 4.1 s in the simulation and 6.2 s in the experiment. The crane lifts the payload from an initial length (1 m) of cable to the desired length (0.7 m) of cable after 4.2 s.

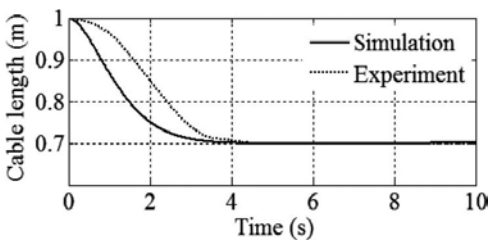

Figure 8. Cargo hoisting motion.

http://dx.doi.org/10.5772/64739

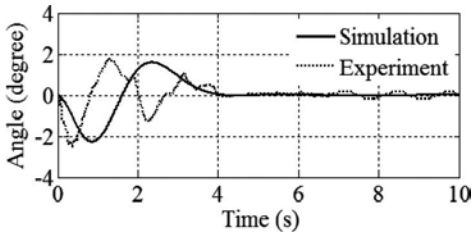

Figure 9. Cargo swing angle ϕ.

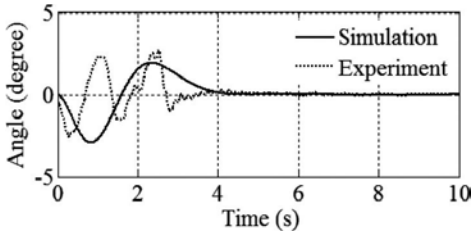

Figure 10. Cargo swing angle θ.

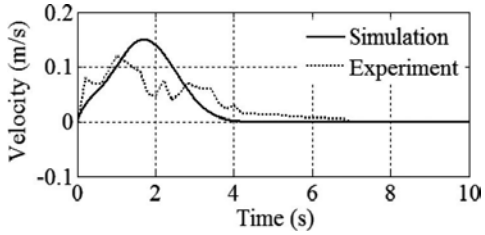

Figure 11. Velocity of bridge motion.

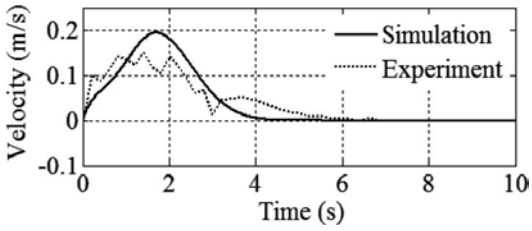

Figure 12. Velocity of trolley motion.

Figures 9 and **10** indicate the responses of the cargo swings. The payload swing angles are in a small boundary during the payload transportation: ϕ_{max} = 2.2° and θ_{max} = 2.9° for the simulation and ϕ_{max} = 2.3° and θ_{max} = 2.4° for the experiment. The simulated cargo swings are

completely vanished after short settling periods, t_s = 4 s for ϕ and t_s = 4.5 s for θ, within one vibration period. Slight steady-state errors remain in the experimental responses, which achieve the approximate steady state after over two oscillation periods.

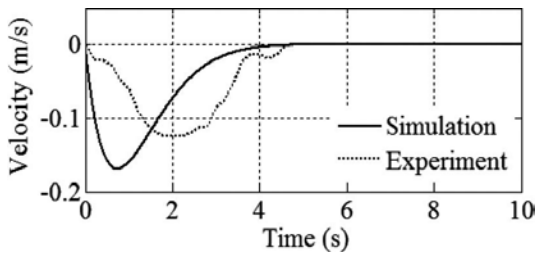

Figure 13. Cargo hoisting velocity.

The velocity components depicted in **Figures 11–15** asymptotically approach to zero. The movements of the bridge and the trolley, as well as the lifting movement of the payload at transient states, composed of two phases, namely, the increasing and decreasing velocity periods. As indicated clearly in the simulated curves, the trolley speeds up within the first 1.7 s and slows down within the last 2.4 s. The cargo is then lifted with increasing speed within the first 0.7 s and with decreasing speed within the remaining 3.5 s.

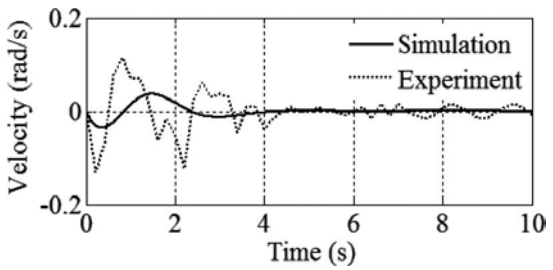

Figure 14. Payload swing velocity $\dot{\varphi}$.

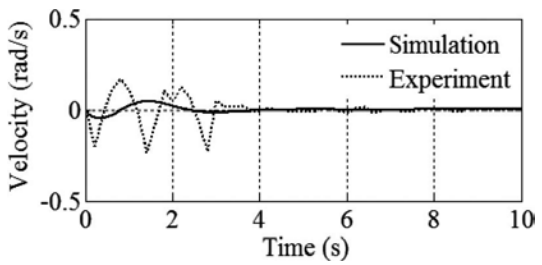

Figure 15. Payload swing velocity $\dot{\theta}$.

http://dx.doi.org/10.5772/64739

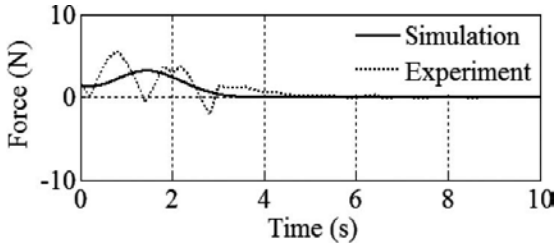

Figure 16. Bridge moving force.

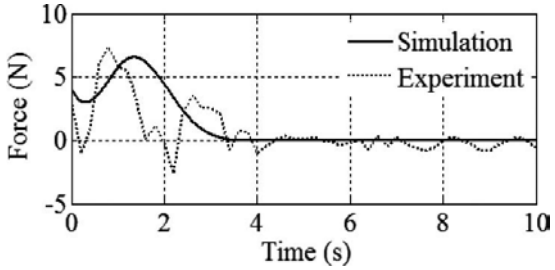

Figure 17. Trolley driving force.

The nonlinear control forces are illustrated in **Figures 16–18**. The simulation responses achieve steady states after 4, 4.1, and 4.2 s for the bridge moving, trolley moving, and cargo lifting forces, respectively.

At steady states, $u_t^{ss} = u_b^{ss} = 0$ N and $u_l^{ss} = -m_c g = -9.81 \times 0.85 = -8.34$ N.

Evidently, differences in responses still exist between the simulation and the experiment responses because the dynamic model and the realistic overhead crane do not match completely. Several nonlinearities that exist in practice, such as the cable flexibility, the backlash of the gear motors, and nonlinear frictions, are not considered in the system dynamics. If the mathematical model is close to a realistic system, then the results will certainly be accurate.

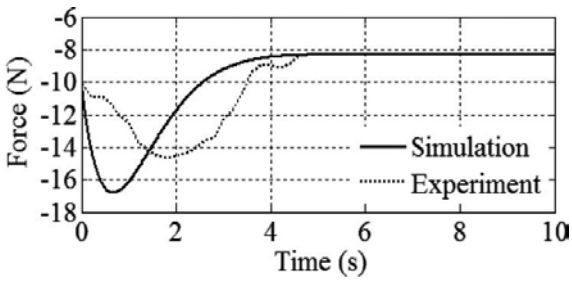

Figure 18. Payload hoisting force.

6. Conclusions

The feedback linearization method provides an effective design tool for controlling nonlinear systems. We improved this technique for application to a class of underactuated mechanical systems. We provided two examples to illustrate the proposed method in which PFL was successfully applied to construct nonlinear controllers for a moving inverted double pendulum and a 3D overhead crane. In general, a nonlinear feedback controller for an underactuated mechanical system consists of two components. The first is for canceling the nonlinearities in the system and the second is for stabilizing state variables.

Acknowledgements

This study was partly supported by the MSIP(Ministry of Science, ICT and Future Planning), Korea, under the Global IT Talent support program (NIPA-2014-ITAH0905140110020001000100100) supervised by the NIPA (National IT Industry Promotion Agency), the Senior-friendly Product R&D program funded by the Ministry of Health & Welfare through the Korea Health Industry Development Institute (KHIDI) (HI15C1027), and the Technology Innovation Program MKE/KEIT (Grant No. 10041629, Implementation of Technologies for Identification, Behavior, and Location of Human based on Sensor Network Fusion Program).

Author details

Le Anh Tuan[1] and Soon-Geul Lee[2*]

*Address all correspondence to: sglee@khu.ac.kr

1 Department of Automotive Engineering, Vietnam Maritime University, Hai Phong, Vietnam

2 Department of Mechanical Engineering, Kyung Hee University, Yongin, Gyeonggi, South Korea

References

[1] E. Lefeber, K.Y. Pettersen, and H. Nijmeijer, "Tracking Control of an Under-actuated Ship," *IEEE Transactions on Control Systems Technology*, vol. 11, no. 1, pp 52–61, 2003.

[2] R. Tedrake, *Under-actuated Robotics*, MIT Open Course Ware (MIT 6.832), Spring 2009.

[3] K. Ogata, *Modern Control Engineering*, Prentice Hall, New Jersey, USA, 2010.

http://dx.doi.org/10.5772/64739

[4] J.-J.E. Slotinne and W. Li, *Applied Nonlinear Control*, Prentice Hall, New Jersey, USA, 1991

[5] H.K. Khalil, *Nonlinear Systems*, Prentice Hall, New Jersey, USA, 2002.

[6] M.W. Spong, "Under-actuated mechanical systems," Book chapter of "Control Problems in Robotics and Automation," B. Siciliano and K.P. Valavanis (Eds), *Lecture Notes in Control and Information Sciences*, Spinger-Verlag, London, Great Britain, 1998.

[7] M.W. Spong, "Partial feedback linearization of under-actuated mechanical systems," *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, Munich, 1994.

[8] L.A. Tuan, S.-G. Lee, V.-H. Dang, S. Moon, and B.S. Kim, "Partial Feedback Linearization Control of a Three Dimensional Overhead Crane," *International Journal of Control, Automation and Systems*, vol.11, no. 4, pp. 718-727, 2013.

Chapter 11

Nonlinear Cascade-Based Control for a Twin Rotor MIMO System

Lidia María Belmonte, Rafael Morales,
Antonio Fernández-Caballero and
José Andrés Somolinos

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64875

Abstract

This research is focused on the development of a nonlinear cascade-based control algorithm for a laboratory helicopter-denominated Twin Rotor MIMO System (TRMS). The TRMS is an underactuated nonlinear multivariable system, characterised by a coupling effect between the dynamics of the propellers and the body structure, which is caused by the action-reaction principle originated in the acceleration and deceleration of the propeller groups. Firstly, this work introduces an extensive description of the platform's dynamics, which was carried out by splitting the system into its electrical and mechanical parts. Secondly, we present a design of a nonlinear cascade-based control algorithm that locally guarantees an asymptotically and exponentially stable behaviour of the controlled generalised coordinates of the TRMS. Lastly, a demonstration of the effectiveness of the proposed approach is provided by means of numerical simulations performed under the MATLAB®/Simulink® environment.

Keywords: nonlinear control, timescale modelling, twin rotor, MIMO systems, laboratory platform

1. Introduction

Currently, there are many possible uses for unmanned aerial vehicles (UAVs), such as inspection operation, battle field operation, forest fire detection, meteorological observation, or search and rescue operation, among others. All these applications require achieving precise control systems. This has motivated an increased interest in the last years from researchers in

developing effective control algorithms for UAVs [1–4]. In many cases, the development of new control strategies requires the use of software and platforms which are able to simulate the operation of the UAVs in order to perform experimental tests for evaluating the different designs. The use of this kind of tools increases the productivity and reduces the development time. For this purpose, different laboratory test rigs have been specifically designed for teaching and research in flight dynamics and control. One such platform is the laboratory helicopter used in this research, namely the Twin Rotor MIMO System (TRMS) [5]. The TRMS is a nonlinear, multivariable and underactuated system, characterised by a coupling effect between the dynamics of the propellers and the body structure, which is caused by the action-reaction principle originated in acceleration and deceleration of the motor-propeller groups. All these features make the control of the TRMS to be perceived as a challenging engineering problem (note that the TRMS, and other laboratory platforms with similar dynamics are more difficult to control than a real helicopter platform [6]). The achievement of an accurate system dynamics model is a challenging problem, whilst, at the same time, an important issue is to develop accurate and efficient control systems.

The development of the dynamic model for the TRMS has been studied by an important number of researches. Ahmad et al. presented mathematical models for the dynamic characterisation of the TRMS, using a black box system identification technique [7] and radial basis function (RBF) networks [8]. Shaheed modelled the dynamics of the TRMS by means of a nonlinear autoregressive process through external input (NARX) approach with a feed-forward neural work and a resilient propagation (RPROP) algorithm [9]. Rahideh and Shaheed have also contributed to the study of the TRMS dynamics by using both Newton- and Lagrange-based methods [10], and two models based on neural networks using Levenberg-Marquardt (LM) and gradient descent (GD) algorithms [11]. Toha and Tokhi presented an adaptive neuro-fuzzy inference system (ANFIS) network design, which was deployed and used for the TRMS modelling [12]. Finally, Tastemirov et al. developed a complete dynamic TRMS model using the Euler-Lagrange method [13].

On the other hand, the design of the control system for the TRMS has been widely discussed through several investigations. Ahmad et al. developed the dynamic model and implemented a feed-forward/open-loop control [14] and a linear quadratic Gaussian control [15]. López-Martínez et al. studied the design of a longitudinal controller based on Lyapunov functions [16], and the application of a nonlinear L_2 controller [17]. Rahideh et al. presented an experimental implementation of an adaptive dynamic nonlinear model inversion control law using artificial neural networks [18]. Other interesting works are those of Tao et al. who designed a parallel distributed fuzzy linear quadratic regulator (LQR) controller [19]. Studies of Reynoso-Meza et al. developed a holistic multi-objective optimisation design technique for controller tuning [20], or the use of a particle swarm optimisation (PSO) algorithm for the proportional-integral-derivative (PID) controller optimisation developed by Coelho et al. [21].

The aim of the present research is to develop a nonlinear cascade-based control algorithm in order to locally guarantee an asymptotically and exponentially stable behaviour of the controlled generalised coordinates of the TRMS. Additionally, the effectiveness of the proposed

http://dx.doi.org/10.5772/64875

nonlinear feedback controller in terms of stabilisation and position tracking performance is demonstrated by means of numerical simulations. Finally, the paper is organised as follows.

Section 2 introduces a description of the TRMS platform by illustrating the details of the dynamics model obtained into two phases: electrical and mechanical parts. Section 3 describes the nonlinear cascade-based controller scheme proposed. The results of the numerical simulations performed under the MATLAB®/Simulink® environment are depicted in Section 4, and, finally, Section 5 is devoted to the conclusions of the work.

2. System description

The TRMS (see **Figure 1**) is a laboratory helicopter platform manufactured by *Feedback Instruments Ltd*©. The TRMS is composed of two propellers that are perpendicular to each other and placed in the extreme of a beam that can rotate freely in both vertical and horizontal planes. Each propeller is driven by a DC motor, thus forming the main and tail rotor of the platform. A main feature of the TRMS is that its movement, unlike a real helicopter, is not achieved by varying the angle of attack of the blades. In this case, the movement of the platform is gotten by means of the variation in the angular velocity of each propeller, which is caused by the change in the control input voltage of each motor.

Figure 1. Twin rotor MIMO system.

This constructive simplification in the TRMS model substantially complicates the dynamics of the system, because a coupling effect between rotors dynamics and the body of the model appears. This effect is caused by the action-reaction principle originated in acceleration and deceleration of the motor-propeller groups.

In addition, the TRMS is an underactuated system. This implies that the number of variables that act as control inputs (voltages applied to the main and the tail rotor; u_m and u_t respectively) is lower than the number of degrees of freedom (DoF) of the system. The DoF are: the pitch (ψ) and the yaw (ϕ) angles, both measured by digital encoders, as well as the angular velocities of the rotors (ω_m for the main rotor and $\omega_t$for the tail), both measured by DC tachometers. Finally, we have to remark that the laboratory platform is locked mechanically, so it cannot move more than ± 2.82 rad in the horizontal plane from -1.05 to $+1.22$ rad in the vertical plane [22]. In other words, $-2.82 \text{ rad} \leq \phi \leq +2.82 \text{ rad}$ and $-1.05 \text{ rad} \leq \psi \leq +1.22 \text{ rad}$.

2.1. Dynamic model of the TRMS

The development of an efficient control algorithm requires a model that represents the dynamic behaviour of the platform under study as accurately as possible. In the particular case of the Twin Rotor MIMO System, the modelling has been addressed from several approaches [7–13]. However, not all of them provide a model that represents the entire complex dynamic behaviour of this experimental platform. For instance, models based on identification techniques have difficulties in representing the effects of coupling, which are characteristic in this platform [7], and neuronal networks and learning algorithms allow obtaining accurate models, but limited to a range of input values and frequencies [11]. Based on previous works developed for the dynamic model of this platform [13, 22–24], a detailed dynamic model of the TRMS has been developed by dividing the whole dynamics of the system in their electrical and mechanical parts. This approach allows not only to adequately capture the complex dynamics behaviour of the TRMS but also the development of novel control algorithms based on nested feedback loops that offer a higher performance than classical control schemes. Moreover, the use of the Euler-Lagrange method in the modelling of the mechanical structure of the TRMS allows a higher adjustment with the real control laboratory platform in comparison with other analytical methods based on the Newtonian approach [25]. The dynamic modelling has been developed in two stages and validated by our research group by means of experimental identification trials. It is presented in the following subsections. The first subsection illustrates the dynamic model of the electrical part, and the second depicts the dynamic model of the mechanical part of the system.

2.1.1. Dynamics of the electrical part

The electrical part of the system is formed by the interface circuit and the DC motors of the main and tail rotors. The interface circuit is the internal electrical circuit that adapts the input control voltages, applied in MATLAB®/Simulink® (u_m for the main rotor and u_t for the tail rotor), to the actual voltage value of the DC motors (v_m for the main rotor and v_t for the tail

http://dx.doi.org/10.5772/64875

rotor). This interface can be modelled as a linear relationship [13], obtaining the following result:

$$v_m = k_{u_m} u_m \tag{1}$$

$$v_t = k_{u_t} u_t \tag{2}$$

where k_{u_m} and k_{u_t} denote the constant gains for the main and tail rotors, respectively. With regard to the DC motors, there are two identical permanent magnet motors, one in each rotor of the TRMS, with the only difference of the mechanical loads (the propellers). Bearing in mind that the dynamics of the motor´s current can be neglected [13], the DC motor dynamics for the main rotor and the tail rotor are the following ones:

$$v_m = R_m i_m + k_{v_m} \omega_m \tag{3}$$

$$v_t = R_t i_t + k_{v_t} \omega_t \tag{4}$$

where i_m and i_t are the motor currents (the subscripts m and t mean "main" and "tail"), R_m and R_t represent the motor resistances, and $k_{v_m} \omega_m$ and $k_{v_t} \omega_t$ denote the electromotive forces of each motor (ω_m and ω_t represent the angular velocities of the each motor). On the other hand, the electromechanical balance of the torques acting on each motor is expressed as:

$$I_{m_1} \dot{\omega}_m = k_{t_m} i_m - f_{v_m} \omega_m - C_{Q_m} \omega_m |\omega_m| \tag{5}$$

$$I_{t_1} \dot{\omega}_t = k_{t_t} i_t - f_{v_t} \omega_t - C_{Q_t} \omega_t |\omega_t| \tag{6}$$

being I_{m_1} and I_{t_1} are the moment of the inertia rotors, $k_{t_m} i_m$ and $k_{t_t} i_t$ denote the electromechanical torques generated by the DC motors, $f_{v_m} \omega_m$ and $f_{v_t} \omega_t$ are the friction torques and $C_{Q_m} \omega_m |\omega_m|$ and $C_{Q_t} \omega_t |\omega_t|$ illustrate the aerodynamic torques.

After substituting the expression for the current intensity of the respective motors [obtained from Eqs. (3) and (4)] and the linear relationships for the interface circuit Eqs. (1) and (2), in

Eqs. (5) and (6), and after operating and rearranging terms, the following two equations are yielded for the main and tail rotors of the TRMS:

$$I_{m_1}\dot{\omega}_m = \frac{k_{t_m}}{R_m}k_{u_m}u_m - \left(\frac{k_{t_m}k_{v_m}}{R_m} + f_{v_m}\right)\omega_m - C_{Q_m}\omega_m|\omega_m| \tag{7}$$

$$I_{t_1}\dot{\omega}_t = \frac{k_{t_t}}{R_t}k_{u_t}u_t - \left(\frac{k_{t_t}k_{v_t}}{R_t} + f_{v_t}\right)\omega_t - C_{Q_t}\omega_t|\omega_t| \tag{8}$$

The dynamics of the electrical part of the TRMS is now expressed in a matrix form, using the following compact notation:

$$\dot{\boldsymbol{\omega}}(t) = \boldsymbol{N}\boldsymbol{u}(t) + \boldsymbol{\Gamma}(\boldsymbol{\omega}(t)) \tag{9}$$

where $\boldsymbol{\omega}(t) = \left[\omega_m, \omega_t\right]^T$ and $\boldsymbol{u}(t) = \left[u_m, u_t\right]^T$ represent the vector of angular velocities and the input control voltages, respectively, and, $\boldsymbol{N}= diag(n_m, \quad n_t)$ and $\boldsymbol{\Gamma}(\boldsymbol{\omega}(t)) = \left[\Gamma_m, \Gamma_t\right]^T$ are defined by:

$$\boldsymbol{N} = \begin{bmatrix} n_m & 0 \\ 0 & n_t \end{bmatrix} = \begin{bmatrix} \dfrac{k_{t_m}k_{u_m}}{I_{m_1}R_m} & 0 \\ 0 & \dfrac{k_{t_t}k_{u_t}}{I_{t_1}R_t} \end{bmatrix} \tag{10}$$

$$\boldsymbol{\Gamma}(\boldsymbol{\omega}(t)) = \begin{bmatrix} \Gamma_m \\ \Gamma_t \end{bmatrix} = \begin{bmatrix} -\left(\dfrac{k_{t_m}k_{v_m}}{R_m} + f_{v_m}\right)\dfrac{\omega_m}{I_{m_1}} - \dfrac{C_{Q_m}}{I_{m_1}}\omega_m|\omega_m| \\ -\left(\dfrac{k_{t_t}k_{v_t}}{R_t} + f_{v_t}\right)\dfrac{\omega_t}{I_{t_1}} - \dfrac{C_{Q_t}}{I_{t_1}}\omega_t|\omega_t| \end{bmatrix} \tag{11}$$

Finally, in order to complete the dynamic model of the electrical part of the TRMS, **Tables 1** and **2** show the parameters used in the model, indicating the description of the parameters, their values and their corresponding units. These values, which are based on the data presented in [13], have been experimentally tuned and validated in the dynamics identification tests that we have performed during our research.

http://dx.doi.org/10.5772/64875

Symbol	Parameter	Value	Units
k_{v_m}	Motor velocity constant	0.0202	V rad^{-1} s
R_m	Motor armature resistance	8	Ω
L_m	Motor armature inductance	0.86×10^{-3}	H
k_{t_m}	Electromagnetic constant torque motor	0.0202	N m A^{-1}
k_{u_m}	Coefficient linear relationship interface circuit	8.5	–
$C_{Q_m}^{+}$	Load factor ($\omega_m \geq 0$)	2.695×10^{-7}	N m s^2 rad^{-2}
$C_{Q_m}^{-}$	Load factor ($\omega_m < 0$)	2.46×10^{-7}	N m s^2 rad^{-2}
f_{v_m}	Viscous friction coefficient	3.89×10^{-6}	N m rad^{-1} s
I_{m_1}	Moment of inertia about the axis of rotation	1.05×10^{-4}	kg m^2

Table 1. Parameters of the main rotor.

Symbol	Parameter	Value	Units
k_{v_t}	Motor velocity constant	0.0202	V rad^{-1} s
R_t	Motor armature resistance	8	Ω
L_t	Motor armature inductance	0.86×10^{-3}	H
k_{t_t}	Electromagnetic constant torque motor	0.0202	N m A^{-1}
k_{u_t}	Coefficient linear relationship interface circuit	6.5	–
C_{Q_t}	Load factor	1.164×10^{-8}	N m s^2 rad^{-2}
f_{v_t}	Viscous friction coefficient	1.715×10^{-6}	N m rad^{-1} s
I_{t_1}	Moment of inertia about the axis of rotation	2.1×10^{-5}	kg m^2

Table 2. Parameters of the tail rotor.

2.1.2. Dynamics of the mechanical part

In the development of the dynamic model of the mechanical part, we consider the mechanics of the TRMS as an assembly of the following three components explained next. The first component is formed by the two rotors, their shields and the free-free beam that links together both rotors. The second component consists in the counterbalance and counterweight beam,

and finally, the third component is the pivoted beam. **Figure 2** helps to clarify the different components considered in the dynamics of the mechanical part of the system. From the previous division, and bearing in mind the notation used in **Figures 3** and **4**, the development of the dynamic model is achieved by means of the application of the Euler-Lagrange formulation. It can be summarised in the following steps:

1. Resolution of the forward kinematics of the three subsystems.
2. Evaluation of the kinetic energy.
3. Evaluation of the potential energy.
4. Obtaining the equations of motion.

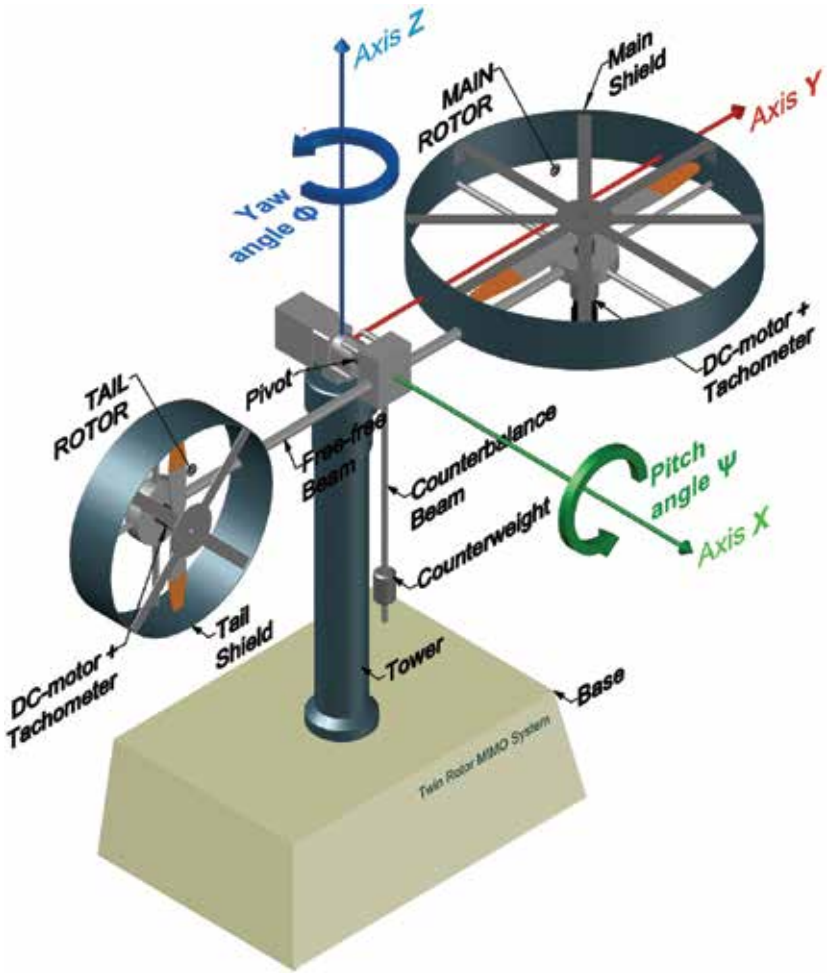

Figure 2. Twin rotor MIMO system (TRMS) prototype platform.

http://dx.doi.org/10.5772/64875

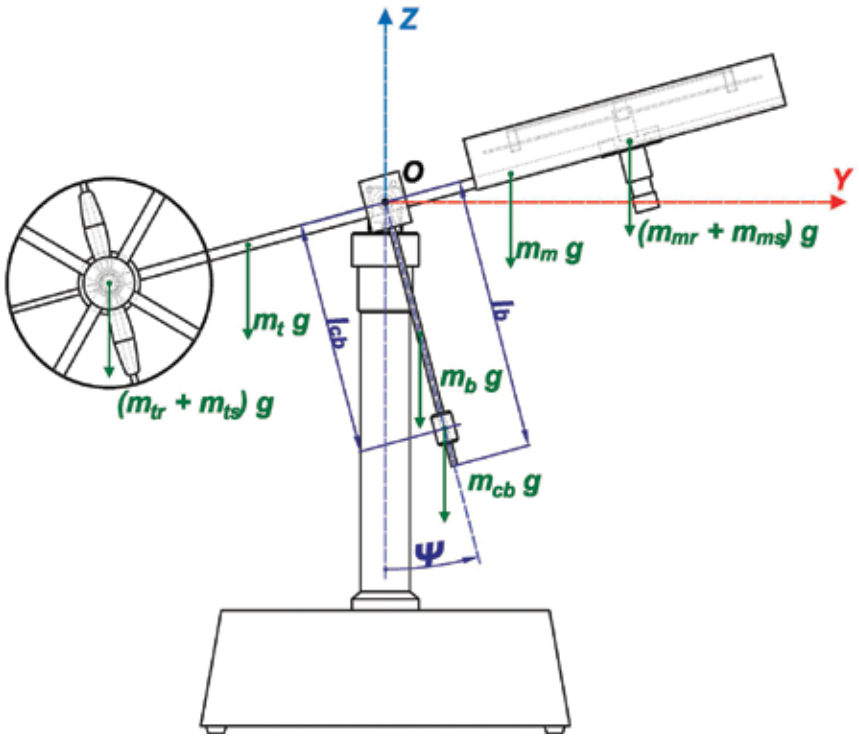

Figure 3. View of the TRMS on a vertical plane.

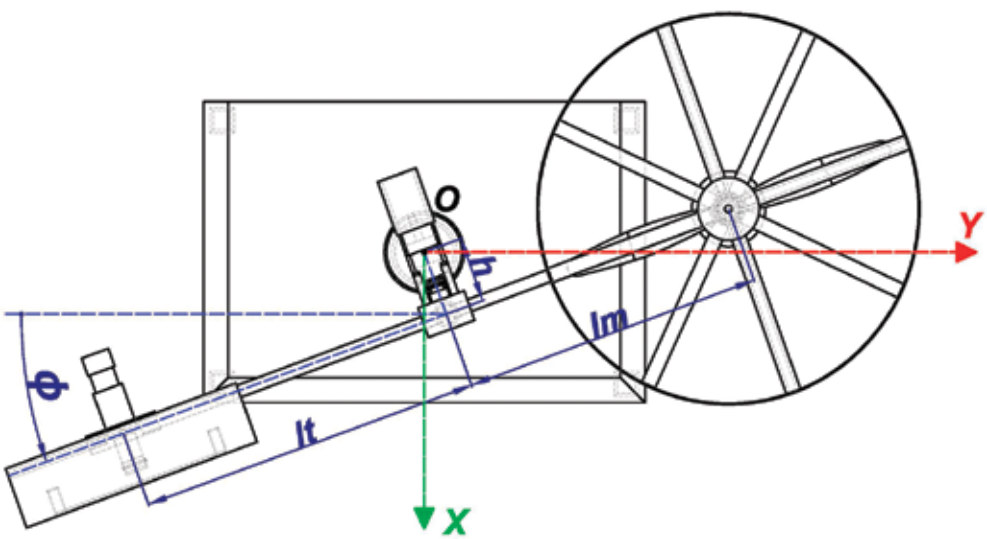

Figure 4. View of the TRMS on a horizontal plane.

2.1.2.1. Resolution of the forward kinematics of the system

The problem of direct kinematics of the TRMS consists in determining the spatial position of the three subsystems considered, according to the reference system located in the upper part of the platform (see **Figures 3** and **4**). Using the Denavit-Hartenberg method, we can express the position of a point on each subsystem (P_1, P_2, P_3) parameterised by R_1, R_2, R_3, which represents the distances between the considerate points and the reference system associated to each subsystem. The results of these positions are expressed in the following three equations (where: $S_\Psi \equiv \sin\Psi$, $C_\Psi \equiv \cos\Psi$, $S_\phi \equiv \sin\phi$ and $C_\phi \equiv \cos\phi$):

$$\boldsymbol{P}_1 = \begin{bmatrix} P_{1_x} & P_{1_y} & P_{1_z} \end{bmatrix}^T = \begin{bmatrix} -R_1 S_\phi C_\psi + hC_\phi & R_1 C_\phi C_\psi + hS_\phi & R_1 S_\psi \end{bmatrix}^T \tag{12}$$

$$\boldsymbol{P}_2 = \begin{bmatrix} P_{2_x} & P_{2_y} & P_{2_z} \end{bmatrix}^T = \begin{bmatrix} -R_2 S_\phi S_\psi + hC_\phi & R_2 C_\phi S_\psi + hS_\phi & -R_2 C_\psi \end{bmatrix}^T \tag{13}$$

$$\boldsymbol{P}_3 = \begin{bmatrix} P_{3_x} & P_{3_y} & P_{3_z} \end{bmatrix}^T = \begin{bmatrix} R_3 C_\phi & R_3 S_\phi & 0 \end{bmatrix}^T \tag{14}$$

2.1.2.2. Evaluation of the kinetic energy

In order to carry out the evaluation of the total kinetic energy of the TRMS, it is necessary to calculate the kinetic energy corresponding to each of the three subsystems previously defined. Starting with the first subsystem, its kinetic energy, T_1, yields:

$$T_1 = \frac{1}{2}\int |v_1|^2 \, dm(R_1) = \frac{1}{2} J_1 \left({C_\psi}^2 \dot{\phi}^2 + \dot{\psi}^2 \right) + \frac{1}{2} h^2 m_{T_1} \dot{\phi}^2 - h S_\psi l_{T_1} m_{T_1} \dot{\phi}\dot{\psi} \tag{15}$$

$$|v_1|^2 = \left({R_1}^2 {C_\psi}^2 + h^2 \right) \dot{\phi}^2 + {R_1}^2 \dot{\psi}^2 - 2 R_1 h S_\psi \dot{\phi}\dot{\psi} \tag{16}$$

where ψ and ϕ represent the yaw and the pitch angle, respectively, and m_{T_1}, l_{T_1}, and J_1 are obtained from the following expressions:

$$\int dm(R_1) = m_{\mathrm{m}} + m_{\mathrm{mr}} + m_{\mathrm{ms}} + m_{\mathrm{t}} + m_{\mathrm{tr}} + m_{\mathrm{ts}} = m_{T_1} \tag{17}$$

$$l_{T_1} = \frac{\int R_1 dm(R_1)}{\int dm(R_1)} = \frac{\left(\frac{m_{\mathrm{t}}}{2} + m_{\mathrm{tr}} + m_{\mathrm{ts}} \right) l_t - \left(\frac{m_{\mathrm{m}}}{2} + m_{\mathrm{mr}} + m_{\mathrm{ms}} \right) l_m}{m_{T_1}} \tag{18}$$

$$J_1 = \left(\frac{1}{3} m_{\mathrm{t}} + m_{\mathrm{tr}} + m_{\mathrm{ts}} \right) {l_t}^2 + \left(\frac{1}{3} m_{\mathrm{m}} + m_{\mathrm{mr}} + m_{\mathrm{ms}} \right) {l_m}^2 + m_{\mathrm{ts}} {r_{\mathrm{ts}}}^2 + \frac{1}{2} m_{\mathrm{ms}} {r_{\mathrm{ms}}}^2 \tag{19}$$

http://dx.doi.org/10.5772/64875

On the other hand, the kinetic energy for the second subsystem, T_2, results in:

$$T_2 = \frac{1}{2}\int |v_2|^2 \, dm(R_2) = \frac{1}{2} J_2 \left({S_\psi}^2 \dot{\phi}^2 + \dot{\psi}^2 \right) + \frac{1}{2} h^2 m_{T_2} \dot{\phi}^2 + h C_\psi l_{T_2} m_{T_2} \dot{\phi}\dot{\psi} \tag{20}$$

$$|v_2|^2 = \left({R_2}^2 {S_\psi}^2 + h^2 \right) \dot{\phi}^2 + {R_2}^2 \dot{\psi}^2 + 2 R_2 h C_\psi \dot{\phi}\dot{\psi} \tag{21}$$

in which the terms m_{T_2}, l_{T_2} and J_2 are the following:

$$\int dm(R_2) = m_{\mathrm{b}} + m_{\mathrm{cb}} = m_{T_2} \tag{22}$$

$$l_{T_2} = \frac{\int R_2 dm(R_2)}{\int dm(R_2)} = \frac{m_{\mathrm{b}} \frac{l_b}{2} + m_{\mathrm{cb}} l_{mcb}}{m_{T_2}} \tag{23}$$

$$J_2 = \frac{1}{3} m_{\mathrm{b}} {l_b}^2 + m_{\mathrm{cb}} {l_{cb}}^2 \tag{24}$$

On the other hand, the kinetic energy for the third subsystem, T_3, gives the following result:

$$T_3 = \frac{1}{2}\int |v_3|^2 \, dm(R_3) = \frac{1}{2} J_3 \dot{\phi}^2 \tag{25}$$

$$|v_3|^2 = {R_3}^2 \dot{\phi}^2 \tag{26}$$

being $J_3 = \frac{1}{3} m_h {l_h}^2$.

Finally, the total kinetic energy of the TRMS, T, is obtained as the sum of the kinetic energy of each subsystem (Eqs. (15), (20) and (25)). One obtains the following result:

$$\begin{aligned} T = T_1 + T_2 + T_3 = {} & \frac{1}{2}\left(J_1 {C_\psi}^2 + J_2 {S_\psi}^2 + J_3 + h^2 \left(m_{T_1} + m_{T_2} \right) \right) \dot{\phi}^2 \\ & + \frac{1}{2}\left(J_1 + J_2 \right) \dot{\psi}^2 + h\left(l_{T_2} m_{T_2} C_\psi - l_{T_1} m_{T_1} S_\psi \right) \dot{\phi}\dot{\psi} \end{aligned} \tag{27}$$

2.1.2.3. Evaluation of the potential energy

Following a similar procedure to the one used in the computation of the kinetic energy, the total potential energy of the TRMS, V, consists of the sum of the potential energy of each of the three subsystems, the free-free beam (including rotors and shields), the counterbalance beam and the pivoted beam. The following result is obtained:

$$V = V_1 + V_2 + V_3 = g\left(S_\psi l_{T_1} m_{T_1} - C_\psi l_{T_2} m_{T_2}\right) \tag{28}$$

where:

$$V_1 = g\int r_{z_1}\left(R_1\right) dm\left(R_1\right) = g\int P_{1_z}\, dm\left(R_1\right) = gS_\psi l_{T_1} m_{T_1} \tag{29}$$

$$V_2 = g\int r_{z_2}\left(R_2\right) dm\left(R_2\right) = g\int P_{2_z}\, dm\left(R_2\right) = -gC_\psi l_{T_2} m_{T_2} \tag{30}$$

$$V_3 = g\int r_{z_3}\left(R_3\right) dm\left(R_3\right) = g\int P_{3_z}\, dm\left(R_3\right) = 0 \tag{31}$$

2.1.2.4. Equations of motion of the TRMS

The last step in the mechanical dynamic model of the TRMS is obtaining the equations of motion of the system. The first step is the computation of the Lagrangian of the system, defined as the difference between the total kinetic energy, defined in Eq. (27), and the total potential energy, defined in Eq. (28), yielding the following:

$$\begin{aligned} L = T - V = {} & \frac{1}{2}\left(J_1 {C_\psi}^2 + J_2 {S_\psi}^2 + J_3 + h^2\left(m_{T_1} + m_{T_2}\right)\right)\dot{\phi}^2 \\ & + \frac{1}{2}\left(J_1 + J_2\right)\dot{\psi}^2 + h\left(l_{T_2} m_{T_2} C_\psi - l_{T_1} m_{T_1} S_\psi\right)\dot{\phi}\dot{\psi} - g\left(S_\psi l_{T_1} m_{T_1} - C_\psi l_{T_2} m_{T_2}\right) \end{aligned} \tag{32}$$

Once the Lagrangian function has been obtained, the equations of motion of the TRMS can be derived using Lagrange's formulation:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\psi}}\right) - \frac{\partial L}{\partial \psi} = \sum M_{iv} \tag{33}$$

http://dx.doi.org/10.5772/64875

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right)-\frac{\partial L}{\partial \phi}=\sum M_{ih} \tag{34}$$

where $\sum \boldsymbol{M_{iv}}$ and $\sum \boldsymbol{M_{ih}}$ represent the sum of the torques of the external forces along the vertical and horizontal axes, respectively. The following expressions illustrate several partial results necessary to achieve the equations of motion represented by Eqs. (33) and (34):

$$\frac{\partial L}{\partial \dot{\psi}}=\left(J_1+J_2\right)\dot{\psi}+h\left(l_{T_2}m_{T_2}C_{\psi}-l_{T_1}m_{T_1}S_{\psi}\right)\dot{\phi} \tag{35}$$

$$\frac{\partial L}{\partial \psi}=\left(\left(J_2-J_1\right)C_{\psi}S_{\psi}\right)\dot{\phi}^2-h\left(l_{T_1}m_{T_1}C_{\psi}+l_{T_2}m_{T_2}S_{\psi}\right)\dot{\phi}\dot{\psi}-g\left(l_{T_1}m_{T_1}C_{\psi}+l_{T_2}m_{T_2}S_{\psi}\right) \tag{36}$$

$$\frac{\partial L}{\partial \dot{\phi}}=\left(J_1{C_{\psi}}^2+J_2{S_{\psi}}^2+J_3+h^2\left(m_{T_1}+m_{T_2}\right)\right)\dot{\phi}+h\left(l_{T_2}m_{T_2}C_{\psi}-l_{T_1}m_{T_1}S_{\psi}\right)\dot{\psi} \tag{37}$$

$$\frac{\partial L}{\partial \phi}=0 \tag{38}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\psi}}\right)=h\left(l_{T_2}m_{T_2}C_{\psi}-l_{T_1}m_{T_1}S_{\psi}\right)\ddot{\phi}+\left(J_1+J_2\right)\ddot{\psi}-h\left(l_{T_1}m_{T_1}C_{\psi}+l_{T_2}m_{T_2}S_{\psi}\right)\dot{\phi}\dot{\psi} \tag{39}$$

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right)=&\left(J_1{C_{\psi}}^2+J_2{S_{\psi}}^2+J_3+h^2\left(m_{T_1}+m_{T_2}\right)\right)\ddot{\phi}-h\left(l_{T_1}m_{T_1}S_{\psi}-l_{T_2}m_{T_2}C_{\psi}\right)\ddot{\psi}\\&-h\left(l_{T_1}m_{T_1}C_{\psi}+l_{T_2}m_{T_2}S_{\psi}\right)\dot{\psi}^2+2\left(\left(J_2-J_1\right)C_{\psi}S_{\psi}\right)\dot{\phi}\dot{\psi}\end{aligned} \tag{40}$$

The sum of the external torques in the vertical axis is shown next:

$$\begin{aligned}&\sum M_{iv}=M_{T_m}-M_{R_t}-M_{F_{\psi}}+M_{I_t}\\&\sum M_{iv}=C_{T_m}\omega_m\left|\omega_m\right|l_m-C_{R_t}\omega_t\left|\omega_t\right|-\left(f_{v_{\psi}}\dot{\psi}+f_{c_{\psi}}sgn\left(\dot{\psi}\right)\right)+k_t\dot{\omega}_t\end{aligned} \tag{41}$$

where $M_{T_m}=C_{T_m}\omega_m\left|\omega_m\right|l_m$ expresses the aerodynamic thrust torque caused by the rotation of the main propeller, $M_{R_t}=C_{R_t}\omega_t\left|\omega_t\right|$ denotes the load torque created by air resistance in the tail rotor, $M_{F_{\psi}}=\left(f_{v_{\psi}}\dot{\psi}+f_{c_{\psi}}sgn\left(\dot{\psi}\right)\right)$ represents the load torque as a result of the friction (including the viscous effects and the Coulomb friction), and $M_{I_t}=k_t\dot{\omega}_t$ represents the inertial counter torque that is caused by the reaction produced by a change in the rotational speed of the tail rotor.

On the other hand, the sum of the external torques in the horizontal axis is as follows:

$$\begin{aligned}\sum M_{ih} &= M_{T_t} - M_{R_m} - M_{F_\phi} - M_c + M_{I_m} \\ \sum M_{ih} &= C_{T_T}\omega_t|\omega_t|l_t C_\psi - C_{R_m}\omega_m|\omega_m|C_\psi \\ &\quad -\left(f_{v_\phi}\dot{\phi} + f_{c_\phi} sgn\left(\dot{\phi}\right)\right) - C_c\left(\phi - \phi_0\right) + k_m\dot{\omega}_m C_\psi\end{aligned} \tag{42}$$

where $M_{T_t} = C_{T_T}\omega_t|\omega_t|l_t C_\psi$ expresses the aerodynamic thrust torque of the tail propeller, $M_{R_m} = C_{R_m}\omega_m|\omega_m|C_\psi$ represents the load torque created by air resistance in the main rotor, $M_{F_\phi} = (f_{v_\phi}\dot{\phi} + f_{c_\phi} sgn(\dot{\phi}))$ denotes the load torque as a result of the friction (including the viscous effects and the Coulomb friction), $M_c = C_c(\phi - \phi_0)$ is the magnitude of torque exerted by the cable (it has a certain stiffness that allows to model it as a spring)), and finally $M_{I_m} = k_m\dot{\omega}_m C_\psi$ represents the inertial counter torque that is caused by the reaction produced by a change in the rotational speed of the main rotor.

Upon merging Eq. (33) to Eq. (42), and after performing some rearrangements, one obtains the following result for the equations of motion:

$$\begin{aligned}&\left(J_1 + J_2\right)\ddot{\psi} + h\left(l_{T_2} m_{T_2} C_\psi - l_{T_1} m_{T_1} S_\psi\right)\ddot{\phi} + \left(\frac{\left(J_1 - J_2\right)}{2} S_{2\psi}\right)\dot{\phi}^2 + g\left(l_{T_1} m_{T_1} C_\psi + l_{T_2} m_{T_2} S_\psi\right) \\ &= C_{T_m}\omega_m|\omega_m|l_m - C_{R_t}\omega_t|\omega_t| - \left(f_{v_\psi}\dot{\psi} + f_{c_\psi} sgn\left(\dot{\psi}\right)\right) + k_t\dot{\omega}_t\end{aligned} \tag{43}$$

$$\begin{aligned}&h\left(l_{T_2} m_{T_2} C_\psi - l_{T_1} m_{T_1} S_\psi\right)\ddot{\psi} + \left(J_1 C_\psi{}^2 + J_2 S_\psi{}^2 + J_3 + h^2\left(m_{T_1} + m_{T_2}\right)\right)\ddot{\phi} - \\ &h\left(l_{T_1} m_{T_1} C_\psi + l_{T_2} m_{T_2} S_\psi\right)\dot{\psi}^2 + \left(\left(J_2 - J_1\right) S_{2\psi}\right)\dot{\phi}\dot{\psi} \\ &= C_{T_T}\omega_t|\omega_t|l_t C_\psi - C_{R_m}\omega_m|\omega_m|C_\psi - \left(f_{v_\phi}\dot{\phi} + f_{c_\phi} sgn\left(\dot{\phi}\right)\right) - C_c\left(\phi - \phi_0\right) + k_m\dot{\omega}_m C_\psi\end{aligned} \tag{44}$$

If we use matrix notation, the dynamic model of the mechanical part of the TRMS can be expressed in a compact form:

$$\boldsymbol{M}\left(\boldsymbol{q}(t)\right)\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}\left(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right)\dot{\boldsymbol{q}}(t) + \boldsymbol{\eta}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \dot{\boldsymbol{\omega}}(t)) = \boldsymbol{E}\left(\boldsymbol{q}(t)\right)\boldsymbol{\Omega}(t) \tag{45}$$

in which $\boldsymbol{q}(t) = [\psi(t), \ \phi(t)]^T$ is the vector of generalised coordinates of the TRMS, $\boldsymbol{\omega}(t) = \left[\omega_m(t), \ \omega_t(t)\right]^T$ is the angular velocity vector, and the matrices $\boldsymbol{M}(\boldsymbol{q}(t))$, $\boldsymbol{C}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))$, $\boldsymbol{E}(\boldsymbol{q}(t))$, and the vectors $\boldsymbol{\Omega}(t)$ and $\boldsymbol{\eta}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \dot{\boldsymbol{\omega}}(t))$ are given by:

http://dx.doi.org/10.5772/64875

$$\boldsymbol{M}\left(\boldsymbol{q}(t)\right)=\begin{bmatrix} J_1+J_2 & h\left(l_{T_2}m_{T_2}C_\psi - l_{T_1}m_{T_1}S_\psi\right) \\ h\left(l_{T_2}m_{T_2}C_\psi - l_{T_1}m_{T_1}S_\psi\right) & J_1{C_\psi}^2 + J_2{S_\psi}^2 + J_3 + h^2\left(m_{T_1}+m_{T_2}\right) \end{bmatrix} \tag{46}$$

$$\boldsymbol{C}\left(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t)\right)=\begin{bmatrix} 0 & \frac{1}{2}\left(J_1-J_2\right)S_{2\psi}\dot{\phi} \\ -h\left(l_{T_1}m_{T_1}C_\psi + l_{T_2}m_{T_2}S_\psi\right)\dot{\psi} & \left(J_2-J_1\right)S_{2\psi}\dot{\psi} \end{bmatrix} \tag{47}$$

$$\boldsymbol{E}\left(\boldsymbol{q}(t)\right)=\begin{bmatrix} C_{T_m}l_m & -C_{R_t} \\ -C_{R_m}C_\psi & C_{T_T}l_tC_\psi \end{bmatrix} \tag{48}$$

$$\boldsymbol{\Omega}(t)=\begin{bmatrix} \omega_m|\omega_m| \\ \omega_t|\omega_t| \end{bmatrix} \tag{49}$$

$$\boldsymbol{\eta}\left(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\dot{\boldsymbol{\omega}}(t)\right)=\boldsymbol{G}\left(\boldsymbol{q}(t)\right)+\boldsymbol{F}\left(\dot{\boldsymbol{q}}(t)\right)+\boldsymbol{T}\left(\boldsymbol{q}(t),\dot{\boldsymbol{\omega}}(t)\right) \tag{50}$$

$$\boldsymbol{G}\left(\boldsymbol{q}(t)\right)=\begin{bmatrix} g\left(l_{T_1}m_{T_1}C_\psi + l_{T_2}m_{T_2}S_\psi\right) \\ 0 \end{bmatrix} \tag{51}$$

$$\boldsymbol{F}\left(\dot{\boldsymbol{q}}(t)\right)=\boldsymbol{F}_v\dot{\boldsymbol{q}}(t)+\boldsymbol{F}_c\left(\dot{\boldsymbol{q}}(t)\right)=\begin{bmatrix} f_{v_\psi} & 0 \\ 0 & f_{v_\phi} \end{bmatrix}\dot{\boldsymbol{q}}(t)+\begin{bmatrix} f_{c_\psi}sgn\left(\dot{\psi}\right) \\ f_{c_\phi}sgn\left(\dot{\phi}\right) \end{bmatrix}=\begin{bmatrix} f_{v_\psi}\dot{\psi}+f_{c_\psi}sgn\left(\dot{\psi}\right) \\ f_{v_\phi}\dot{\phi}+f_{c_\phi}sgn\left(\dot{\phi}\right) \end{bmatrix} \tag{52}$$

$$\begin{aligned} \boldsymbol{T}\left(\boldsymbol{q}(t),\dot{\boldsymbol{\omega}}(t)\right) &= \boldsymbol{M_c}\left(\boldsymbol{q}(t)\right)-\boldsymbol{M_g}\left(\boldsymbol{q}(t)\right)\dot{\boldsymbol{\omega}}(t)=\begin{bmatrix} 0 \\ C_c(\phi-\phi_0) \end{bmatrix}-\begin{bmatrix} 0 & k_t \\ k_mC_\psi & 0 \end{bmatrix}\dot{\boldsymbol{\omega}}(t) \\ &= \begin{bmatrix} -k_t\dot{\omega}_t \\ C_c(\phi-\phi_0)-k_m\dot{\omega}_mC_\psi \end{bmatrix} \end{aligned} \tag{53}$$

Finally, after substituting Eqs. (51)–(53) into Eq. (50), the following yields:

$$\boldsymbol{\eta}\left(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t),\dot{\boldsymbol{\omega}}(t)\right)=\begin{bmatrix} g\left(l_{T_1}m_{T_1}C_\psi + l_{T_2}m_{T_2}S_\psi\right)+f_{v_\psi}\dot{\psi}+f_{c_\psi}sgn\left(\dot{\psi}\right)-k_t\dot{\omega}_t \\ f_{v_\phi}\dot{\phi}+f_{c_\phi}sgn\left(\dot{\phi}\right)+C_c\left(\phi-\phi_0\right)-k_m\dot{\omega}_mC_\psi \end{bmatrix} \tag{54}$$

Symbol	Parameter	Value	Units
l_{t}	Length of the tail part of the free-free beam	0.282	m
l_{m}	Length of the main part of the free-free beam	0.246	m
l_{b}	Length of the counterbalance beam	0.290	m
l_{cb}	Distance between the counterweight and the join	0.276	m
r_{ms}	Radius of the main shield	0.155	m
r_{ts}	Radius of the tail shield	0.1	m
h	Length of the pivoted beam	0.06	m
m_{tr}	Mass of the tail DC motor and tail rotor	0.221	kg
m_{mr}	Mass of the main DC motor and main rotor	0.236	kg
m_{cb}	Mass of the counterweight	0.068	kg
m_{t}	Mass of the tail part of the free-free beam	0.015	kg
m_{m}	Mass of the main part of the free-free beam	0.014	kg
m_{b}	Mass of the counterbalance beam	0.022	kg
m_{ts}	Mass of the tail shield	0.119	kg
m_{ms}	Mass of the main shield	0.219	kg
m_{h}	Mass of pivoted beam	0.01	kg

Table 3. Mechanical parameters.

Symbol	Parameter	Value	Units
$C_{T_m}^{+}$	Thrust torque coefficient of the main rotor ($\omega_m \geq 0$)	1.53×10^{-5}	$\mathrm{N\,s^2\,rad^{-2}}$
$C_{T_m}^{-}$	Thrust torque coefficient of the main rotor ($\omega_m < 0$)	8.8×10^{-6}	$\mathrm{N\,s^2\,rad^{-2}}$
C_{R_t}	Load torque coefficient of the tail rotor	9.7×10^{-8}	$\mathrm{N\,m\,s^2\,rad^{-2}}$
f_{v_ψ}	Viscous friction coefficient	0.0024	$\mathrm{N\,m\,s\,rad^{-1}}$
f_{c_ψ}	Coulomb friction coefficient	5.69×10^{-4}	N m
k_{t}	Coefficient of the inertial counter torque created by the change in ω_t	2.6×10^{-5}	$\mathrm{N\,m\,s^2\,rad^{-1}}$

Table 4. Parameters of the pitch movement.

http://dx.doi.org/10.5772/64875

Symbol	Parameter	Value	Units
$C_{T_t}^{+}$	Thrust torque coefficient of the tail rotor ($\omega_t \geq 0$)	3.25×10^{-6}	N s^2 rad^{-1}
$C_{T_t}^{-}$	Thrust torque coefficient of the tail rotor ($\omega_t < 0$)	1.72×10^{-6}	N s^2 rad^{-2}
$C_{R_m}^{+}$	Load torque coefficient of the main rotor ($\omega_m \geq 0$)	4.9×10^{-7}	N m s^2 rad^{-2}
$C_{R_m}^{-}$	Load torque coefficient of the main rotor ($\omega_m < 0$)	4.1×10^{-7}	N m s^2 rad^{-2}
f_{v_ϕ}	Viscous friction coefficient	0.03	N m s rad^{-1}
f_{c_ϕ}	Coulomb friction coefficient	3×10^{-4}	N m
C_c	Coefficient of the elastic force torque created by the cable	0.016	N m rad^{-1}
ϕ_0	Constant for the calculation of the torque of the cable	0	rad
k_m	Coefficient of the inertial counter torque created by the change in ω_m	2×10^{-4}	N m s^2 rad^{-1}

Table 5. Parameters of the yaw movement.

Finally, in order to complete the dynamic modelling for the mechanical part of the TRMS, **Tables 3–5** show in detail the parameters used in the model. For each parameter, its description, its value and the corresponding units is included. The initial approximation of these values was based in the developments described in [13]. Additionally, some values of the parameters have been tuned by carrying out several identification trials.

3. Design of the control system

In this section, the proposed nonlinear control for the TRMS platform is described. The proposed control is based on the division between the electrical and mechanical dynamics of the system and uses a cascade-type nonlinear control algorithm. **Figure 5** displays the proposed control scheme. As it can be observed, the proposed design is composed of two independent stages (or control loops) that are utilised to achieve stabilisation and precise trajectory tracking tasks for the controlled position of the generalised system coordinates. It should be noted that the proposed solution has been designed to overcome one of the limitations of the TRMS, which is the fact of being an underactuated system. As result of this fact, it only has two control actions (the input voltages of the main and tail rotors) to control the four degrees of freedom of the system (the pitch and yaw angles, and the angular velocities of the propellers). In this way, in order to meet this objective, once the dynamics of the TRMS have been decoupled, a nonlinear multivariable inner loop is closed to control the vector of the

angular velocities, and then, a nonlinear multivariable outer loop is closed to control the vector of the generalised coordinates of the system. This solution, based on a control scheme with two nested loops, allows a simplification in the design procedure as a result of its division into two simpler processes. Moreover, the scheme can be implemented more easily and safely than the standard controllers.

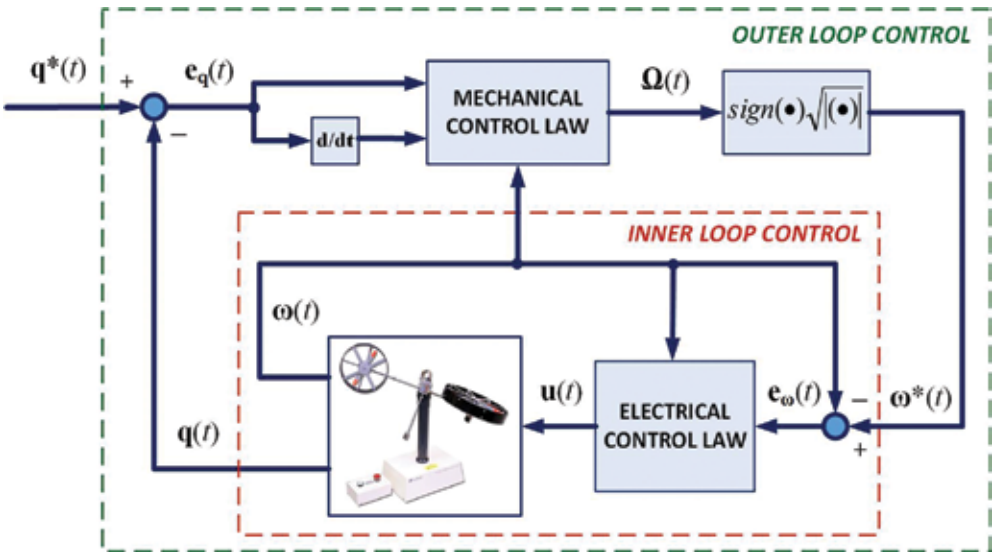

Figure 5. Nonlinear control scheme for the TRMS.

In the following subsections we describe the specifications and objectives of each control loop, defined as the inner loop or electrical controller and the outer loop or mechanical controller.

3.1. Inner loop control

The objective of the inner loop control is to determine the input voltages of the main and tail rotors (simulated in the MATLAB®/Simulink® environment), $\boldsymbol{u}(t) = \left[u_m, u_t\right]^T$, in order to eliminate the difference between the vector of reference angular velocities, $\boldsymbol{\omega}^*(t) = \left[\omega_m^*, \omega_t^*\right]^T$, calculated in the outer loop stage (as will be described in the next subsection), and the current vector of angular velocities of the propellers of the TRMS, $\boldsymbol{\omega}(t) = \left[\omega_m, \omega_t\right]^T$.

The magnitude of the input control voltage vector, $\boldsymbol{u}(t)$, necessary to achieve an asymptotically stable convergent behaviour of the tracking error trajectories, is calculated as the following nonlinear control law:

$$\boldsymbol{u}(t) = \boldsymbol{N}^{-1}\left[\boldsymbol{\gamma}_e(t) - \boldsymbol{\Gamma}\left(\boldsymbol{\omega}(t)\right)\right] \tag{55}$$

http://dx.doi.org/10.5772/64875

where $\boldsymbol{N}$ and $\boldsymbol{\Gamma}(\boldsymbol{\omega}(t))$ where defined in Eqs. (10) and (11), respectively, and $\boldsymbol{\gamma_e}(t) = \left[\gamma_m, \gamma_t\right]^T$ represents a vector of auxiliary control inputs, given by the following expression:

$$\boldsymbol{\gamma}_e(t) = -\boldsymbol{K}_P^e \boldsymbol{e}_{\boldsymbol{\omega}}(t) = -\boldsymbol{K}_P^e[\boldsymbol{\omega}(t) - \boldsymbol{\omega}^*(t)] \tag{56}$$

where $\boldsymbol{K}_P^e \in \mathbb{R}^{2\times 2}$ is a constant diagonal positive definite matrix that represents the design elements of a vector-valued classical proportional controller and $\boldsymbol{e}_{\boldsymbol{\omega}}(t) = \boldsymbol{\omega}(t) - \boldsymbol{\omega}^*(t)$ is the angular velocity error vector, which satisfies the following predominantly linear dynamic:

$$\dot{\boldsymbol{\omega}}(t) + \boldsymbol{K}_P^e \boldsymbol{e}_{\boldsymbol{\omega}}(t) = \boldsymbol{0} \tag{57}$$

Finally, the coefficients of the matrix K_P^e are chosen so as to render the closed-loop characteristic polynomial vectors into a Hurwitz polynomial vector with desirable roots.

3.2. Outer loop control

The aim of the outer loop control (mechanical controller) is to determine the required values for the angular velocities of the two rotors, $\boldsymbol{\omega}^*(t) = \left[\omega_m^*, \omega_t^*\right]^T$, which will be the reference inputs of the electrical loop (described in the above subsection), in order to eliminate the difference between the generalised coordinates of the TRMS, $\boldsymbol{q}(t) = [\psi, \phi]^T$, and the reference trajectories for the generalised coordinates of the TRMS $\boldsymbol{q}^*(t) = [\psi^*, \phi^*]^T$.

As a previous step for determining the mechanical control law, a simplification in the dynamic mechanical modelling of the TRMS has been considered. If we assume that the movement of the platform is sufficiently smooth, the terms of the inertial counter torques, which are caused by the reaction produced by the changes in the rotational speed of each rotor, $M_{I_t} = k_t \dot{\omega}_t$ and $M_{I_m} = k_m \dot{\omega}_m C_\psi$ included in Eqs. (53) and (54), can be considered negligible in comparison with the other terms. In this way, the dynamic equation of the mechanical part of the TRMS can be rewritten as:

$$\boldsymbol{M}\left(\boldsymbol{q}(t)\right)\ddot{\boldsymbol{q}}(t) + \boldsymbol{D}\left(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right) = \boldsymbol{E}\left(\boldsymbol{q}(t)\right)\boldsymbol{\Omega}(t) \tag{58}$$

where the matrices $\boldsymbol{M}(\boldsymbol{q}(t))$, $\boldsymbol{E}(\boldsymbol{q}(t))$, and $\boldsymbol{\Omega}(t)$ were defined in the previous section and the new matrix $\boldsymbol{D}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) = \left[D_\psi, D_\phi\right]^T$ is given by:

$$D_\psi = \frac{1}{2}\left(J_1 - J_2\right) S_{2\psi} \dot{\phi}^2 + g\left(l_{T_1} m_{T_1} C_\psi + l_{T_2} m_{T_2} S_\psi\right) + \left(f_{v_\psi}\dot{\psi} + f_{c_\psi} sgn\left(\dot{\psi}\right)\right) \tag{59}$$

$$D_{\phi} = -h\left(l_{T_1} m_{T_1} C_{\psi} + l_{T_2} m_{T_2} S_{\psi}\right)\dot{\psi}^2 + \left(\left(J_2 - J_1\right) S_{2\psi}\right)\dot{\phi}\dot{\psi} + \left(f_{v_{\phi}}\dot{\phi} + f_{c_{\phi}} sgn\left(\dot{\phi}\right)\right) + C_c\left(\phi - \phi_0\right) \tag{60}$$

The following nonlinear feedback control input vector, $\boldsymbol{\Omega}(t)$, is synthesised as a multivariable proportional-derivative (PD) controller with a cancellation term:

$$\boldsymbol{\Omega}(t) = \boldsymbol{E}^{-1}\left(\boldsymbol{q}(t)\right)\left[\boldsymbol{M}\left(\boldsymbol{q}(t)\right)\boldsymbol{\gamma}_{\boldsymbol{m}}(t) + \boldsymbol{D}\left(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right)\right] \tag{61}$$

where $\boldsymbol{\gamma}_m(t) = \left[\gamma_{\psi}, \gamma_{\phi}\right]^T$ is given by the following expression:

$$\boldsymbol{\gamma}_{\boldsymbol{m}}(t) = \ddot{\boldsymbol{q}}(t) = \ddot{\boldsymbol{q}}^*(t) - \boldsymbol{K}_{\boldsymbol{D}}^{\boldsymbol{m}}\left(\dot{\boldsymbol{q}}(t) - \dot{\boldsymbol{q}}^*(t)\right) - \boldsymbol{K}_{\boldsymbol{P}}^{\boldsymbol{m}}\left(\boldsymbol{q}(t) - \boldsymbol{q}^*(t)\right) \tag{62}$$

in which $\boldsymbol{K}_{\boldsymbol{D}}^{\boldsymbol{m}}$ and $\boldsymbol{K}_{\boldsymbol{P}}^{\boldsymbol{m}} \in \mathbb{R}^{2\times 2}$ are the diagonal positive definite matrices that represent the design elements of a vector-valued classical PD controller. Thereby, for the mechanical part, the closed loop tracking error vector, $\boldsymbol{e}_{\boldsymbol{q}}(t) = \boldsymbol{q}(t) - \boldsymbol{q}^*(t)$, evolves governed by:

$$\ddot{\boldsymbol{e}}_{\boldsymbol{q}}(t) + \boldsymbol{K}_{\boldsymbol{D}}^{\boldsymbol{m}}\dot{\boldsymbol{e}}_{\boldsymbol{q}}(t) + \boldsymbol{K}_{\boldsymbol{P}}^{\boldsymbol{m}}\boldsymbol{e}_{\boldsymbol{q}}(t) = 0 \tag{63}$$

The controller design matrices $\boldsymbol{K}_{\boldsymbol{D}}^{\boldsymbol{m}}$ and $\boldsymbol{K}_{\boldsymbol{P}}^{\boldsymbol{m}}$ have been selected based in the philosophy used for the electrical controller. They must be selected to render closed-loop characteristic polynomial vectors into a Hurwitz polynomial vector with desirable roots. Finally, the necessary angular velocity vector values, $\boldsymbol{\omega}^*(t) = \left[\omega_m^*, \omega_t^*\right]^T$, are obtained from the input control vector, $\boldsymbol{\Omega}(t) = \left[\omega_m|\omega_m|\ \ \omega_t|\omega_t|\right]^T$, by performing the following operation:

$$\boldsymbol{\omega}^*(t) = \begin{bmatrix} \omega_m^* \\ \omega_t^* \end{bmatrix} = \begin{bmatrix} sgn\left(\omega_m|\omega_m|\right)\cdot\sqrt{\left|\omega_m|\omega_m|\right|} \\ sgn\left(\omega_t|\omega_t|\right)\cdot\sqrt{\left|\omega_t|\omega_t|\right|} \end{bmatrix} \tag{64}$$

4. Results

This section describes the numerical simulations carried out in the MATLAB®/Simulink® environment for the sake of verifying the efficiency of the proposed control approach in terms of quick convergence of the tracking errors to a small neighbourhood of zero, smooth transient

http://dx.doi.org/10.5772/64875

responses and low control effort. In the simulations, the desired reference trajectory for the pitch (ψ) and the yaw (ϕ) angles have been defined by the next expression:

$$\boldsymbol{q}^{*}(t)=\begin{bmatrix}\psi^{*}\\ \phi^{*}\end{bmatrix}=\begin{bmatrix}A_{0_\psi}+A_{1_\psi}\left(2\sin\left(\omega_{1_\psi}t\right)+\sin\left(\omega_{2_\psi}t\right)\right)\\ A_{1_\phi}\sin\left(\omega_{1_\phi}t\right)+A_{2_\phi}\left(\sin\left(\omega_{2_\phi}t\right)+\sin\left(\omega_{3_\phi}t\right)\right)\end{bmatrix} \tag{65}$$

where $\boldsymbol{q}^{*}(t)=[\psi^{*}(t),\phi^{*}(t)]^{T}$ is the reference trajectory vector of the generalised coordinates, and the values of the constants used in the above expressions are given by:

$$A_{0_\psi}=0.4\text{ rad};\quad A_{1_\psi}=0.1\text{ rad};\quad A_{1_\phi}=0.8\text{ rad};\quad A_{2_\phi}=0.3\text{ rad}; \tag{66}$$

$$\omega_{1_\psi}=0.0785\text{ rad/s};\quad \omega_{2_\psi}=0.0157\text{ rad/s}; \tag{67}$$

$$\omega_{1_\phi}=0.157\text{ rad/s};\quad \omega_{2_\phi}=0.0785\text{ rad/s};\quad \omega_{3_\phi}=0.0157\text{ rad/s}; \tag{68}$$

On the other hand, the values used in the simulation of the dynamic model of the TRMS, electrical parameters (main and tail rotors), mechanical parameters and dimensional parameters of the platform are detailed in **Tables 1–5**. The initial position of the TRMS has been defined as $\boldsymbol{q}_0(t)=\left[\psi_0,\phi_0\right]^{T}=[0,0]^{T}$ rad, representing a different value of the initial position than the reference trajectory vector. This choice of the starting position has been made to demonstrate the exponential convergence of the desired trajectories. With regard to the controller design parameters, it must be remarked that they have been selected to make the dynamics of the inner loop much faster than the outer loop dynamics, all this in order to ensure the functioning of the cascade controller [26]. The resulting values are as follows:

$$\boldsymbol{K_P^e}=diag(10.5,6.2); \tag{69}$$

$$\boldsymbol{K_D^m}=diag\left(8.20,3.85\right);\quad \boldsymbol{K_P^m}=diag(13.20,2.205); \tag{70}$$

Figures 6 and **7** show the performance of the proposed control scheme. **Figure 6** illustrates a comparative between the desired trajectory, $\boldsymbol{q}^{*}(t)=[\psi^{*}(t),\phi^{*}(t)]^{T}$, and the real trajectory of the TRMS, $\boldsymbol{q}(t)=[\psi(t),\phi(t)]^{T}$. The difference between these trajectories, or, in other words, the error vector of generalised coordinates, $\boldsymbol{e_q}(t)=\boldsymbol{q}(t)-\boldsymbol{q}^{*}(t)=[\psi(t)-\psi^{*}(t),\phi(t)-\phi^{*}(t)]^{T}$, is represented in **Figure 7**. The exponential convergence of the desired trajectories is observed,

with the error bounded to a small neighbourhood to zero, and the robustness against large initial errors.

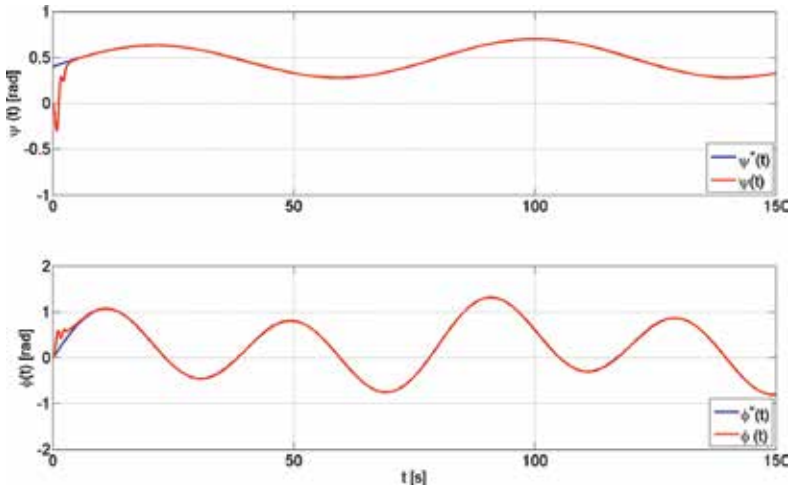

Figure 6. Real and desired evolution of the vector of generalised coordinates of the TRMS, $\boldsymbol{q}(t) = [\psi(t), \phi(t)]^T$.

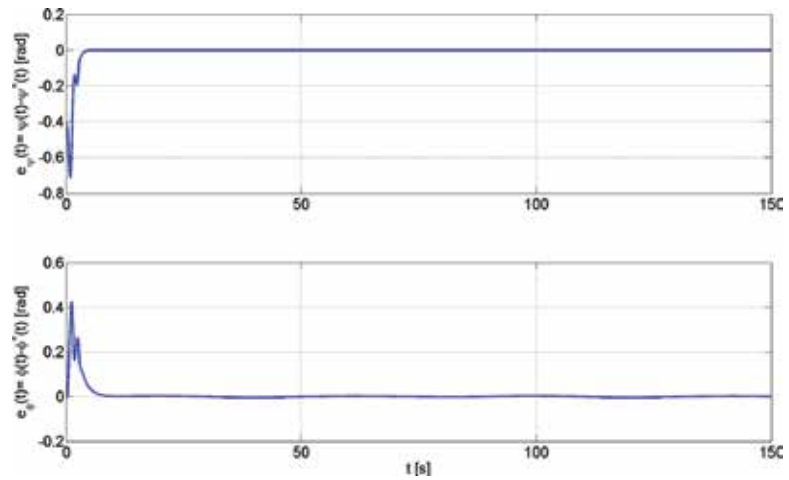

Figure 7. Evolution of the error vector of the generalised coordinates of the TRMS, $\boldsymbol{e_q}(t) = \boldsymbol{q}(t) - \boldsymbol{q}^*(t) = [\psi(t) - \psi^*(t), \phi(t) - \phi^*(t)]^T$.

Another graph that shows the excellent performance of the outer control loop is shown in **Figure 8**, where the auxiliary control input vector of the mechanical proportional-derivative (PD) controller (Eq. (62)) can be observed. This figure shows the quick convergence of the auxiliary control inputs of the mechanical controller to a small value of the origin in the

http://dx.doi.org/10.5772/64875

reference trajectory tracking vector error phase space, $\boldsymbol{e_q}(t)$, in a globally asymptotic exponential dominated manner.

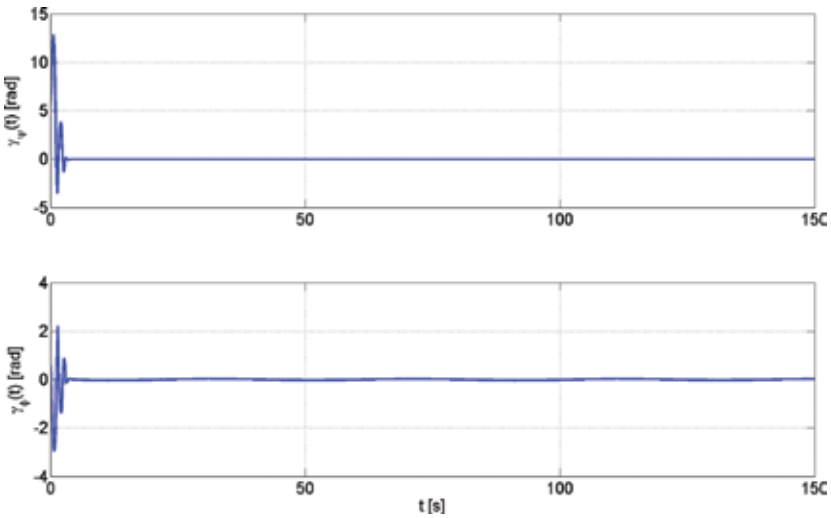

Figure 8. Evolution of the auxiliary control input vector of the mechanical multivariable PD controller, $\boldsymbol{\gamma}_m(t) = \left[\gamma_\psi(t), \gamma_\phi(t)\right]^T$.

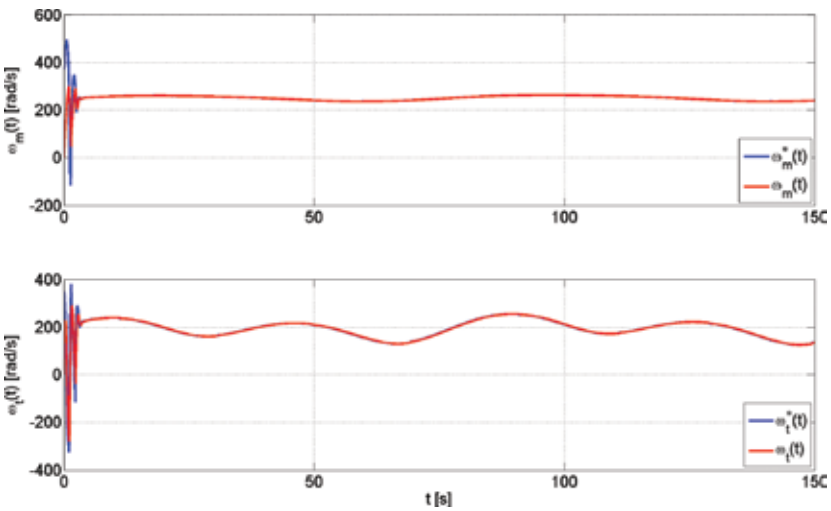

Figure 9. Real and desired evolution trajectories of the angular velocity vector, $\boldsymbol{\omega}^*(t) = \left[\omega_m^*(t),\ \omega_t^*(t)\right]^T$ and $\boldsymbol{\omega}(t) = \left[\omega_m(t),\ \omega_t(t)\right]^T$.

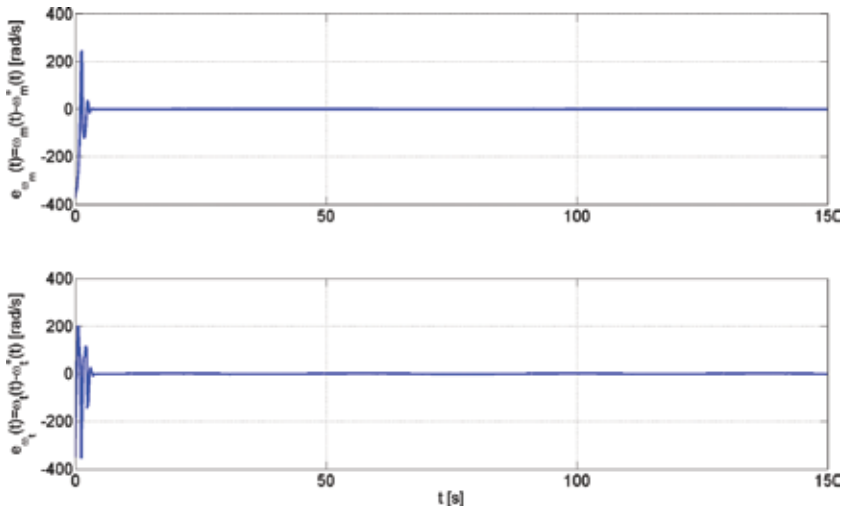

Figure 10. Evolution of the angular velocity error vector, $e_{\omega}(t) = \omega(t) - \omega^*(t) = [\omega_m(t) - \omega_m^*(t), \omega_t(t) - \omega_t^*(t)]\, T$.

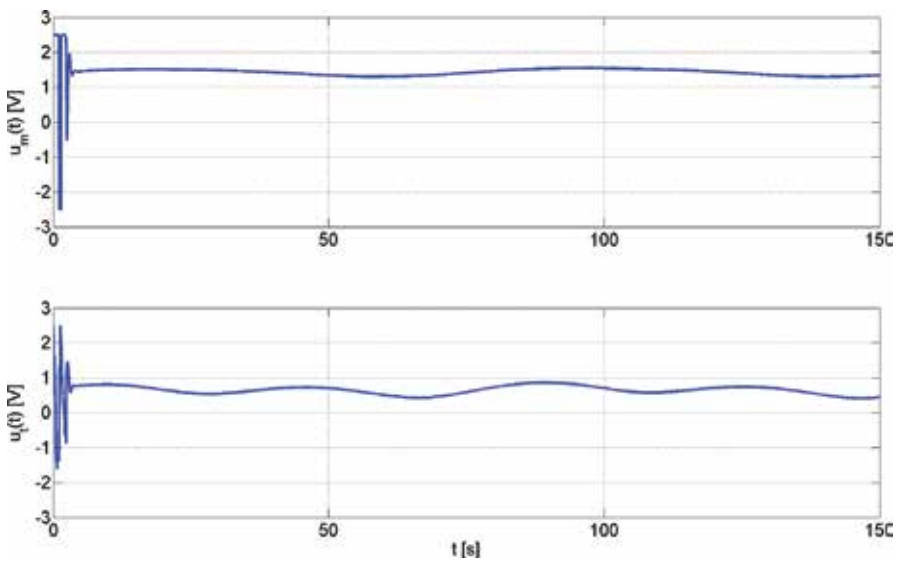

Figure 11. Evolution of the input voltage vector of the TRMS, $\boldsymbol{u}(t) = \left[u_m(t),\ u_t(t)\right]^T$.

On the other hand, the efficiency of the inner loop control (electrical controller) is depicted in **Figure 9**, including a comparative between the reference angular velocity vector, $\boldsymbol{\omega}^*(t) = \left[\omega_m^*(t), \omega_t^*(t)\right]^T$, obtained from the output of the outer loop, and the real magnitudes

http://dx.doi.org/10.5772/64875

of angular velocity vector, $\boldsymbol{\omega}(t) = \left[\omega_m(t), \omega_t(t)\right]^T$. The evolution of the angular velocity error vector, $\boldsymbol{e}_{\boldsymbol{\omega}}(t) = \boldsymbol{\omega}(t) - \boldsymbol{\omega}^*(t) = \left[\omega_m(t) - \omega_m^*(t), \omega_t(t) - \omega_t^*(t)\right]^T$, is also shown in **Figure 10**.

To conclude this section, the input voltages in the MATLAB®/Simulink® environment, $\boldsymbol{u}(t) = \left[u_m(t), u_t(t)\right]^T$, for the main and tail rotors, are represented in **Figure 11**. From these graphs, it can be observed that the proposed control scheme has been realised to avoid saturations on these voltages, which in the simulation MATLAB®/Simulink® environment have been set to ± 2.5 V (similarly to the real prototype platform).

5. Conclusions

In this research, a novel nonlinear cascade-based control has been developed for the TRMS platform. The performance of the controller shows very satisfactory results in terms of convergence of the tracking errors for the generalised coordinates of the TRMS to a small neighbourhood to zero, smooth transient responses, low control efforts and robustness against large initial errors and parametric uncertainties in the model. The proposed control is an important base for the subsequent design of novel robust control algorithms in UAV platforms, which interest is notably increasing in recent years thanks to their multiple possibilities and applications. This will be the topic of our future research.

Acknowledgements

This work has been partially supported by Spanish Ministerio de Economía y Competitividad/FEDER under TEC2016-80986-R, DPI2016-80894-R, TIN2013-47074-C2-1-R and DPI2014-53499-R grants. Lidia M. Belmonte holds an FPU Scholarship (FPU014/05283) from the Spanish Government.

Author details

Lidia María Belmonte[1], Rafael Morales[1*], Antonio Fernández-Caballero[1] and José Andrés Somolinos[2]

*Address all correspondence to: Rafael.Morales@uclm.es

1 University of Castilla-La Mancha, School of Industrial Engineering, Albacete, Spain

2 Polytechnic University of Madrid, School of Naval Engineering, Madrid, Spain

References

[1] Castillo P., Lozano R., Dzul A.E. Modelling and control of mini-flying machines. London: Springer; 2005. doi:10.1007/1-84628-179-2

[2] Raffo G.V., Ortega M.G., Rubio F.R. An integral predictive/nonlinear H_∞ control structure for a quadrotor helicopter. Automatica. 2010; 46(1):29–39. doi:10.1016/j.automatica.2009.10.018

[3] Cai G., Chen B.M., Dong X., Lee T.H. Design and implementation of a robust and nonlinear flight control system for an unmanned helicopter. Mechatronics. 2011; 21(5): 803–820. doi:10.1016/j.mechatronics.2011.02.002

[4] Fernández-Caballero A., Belmonte L.M., Morales R., Somolinos J.A. Generalized proportional integral control for an unmanned quadrotor system. International Journal of Advanced Robotic Systems. 2015; 12(85): 1–14. doi:10.5772/60833

[5] Feedback Co. Twin rotor MIMO system 33-220 user manual. 1998

[6] Mullhaupt P., Srinivasan B., Lévine J., Bonvin D. A toy more difficult to control than the real thing. In: Proceedings of the European Control Conference (ECC'97). Brussels, July 1997

[7] Ahmad S.M., Chipperfield A.J., Tokhi M.O. Parametric modelling and dynamic characterization of a two-degree-of-freedom twin-rotor multi-input multi-output system. Proceedings of the Institution of Mechanical Engineers Part G, Journal of Aerospace Engineering. 2001; 215(2):63–78. doi:10.1243/0954410011531772

[8] Ahmad S.M., Shaheed M.H., Chipperfield A.J., Tokhi M.O. Non-linear modelling of a one-degree-of-freedom twin-rotor multi-input multi-output system using radial basis function networks. Proceedings of the Institution of Mechanical Engineers Part G, Journal of Aerospace Engineering. 2002; 216(4):197–208.doi: 10.1243/09544100260369731

[9] Shaheed M.H. Feedforward neural network based non-linear dynamic modelling of a TRMS using RPROP algorithm. Aircraft Engineering and Aerospace Technology: An International Journal. 2005; 77(1):13–22. doi:10.1108/00022660510576000

[10] Rahideh A., Shaheed M.H. Mathematical dynamic modelling of a twin-rotor multiple input-multiple output system. Proceedings of the Institution of Mechanical Engineers Part I, Journal of Systems and Control Engineering. 2007; 221(1): 89–101. doi: 10.1243/09596518JSCE292

[11] Rahideh A., Shaheed M.H., Huijberts H.J.C Dynamic modelling of a TRMS using analytical and empirical approaches. Control Engineering Practice. 2008; 16(3): 241–259. doi:10.1016/j.conengprac.2007.04.008

[12] Toha S.F., Tokhi M.O. ANFIS modelling of a twin rotor system using particle swarm optimisation and RLS. In: Cybernetic Intelligent Systems (CIS), 2010 IEEE 9th International Conference on; 1–2 Sept. 2010. IEEE. doi:10.1109/UKRICIS.2010.5898130

[13] Tastemirov A., Lecchini-Visintini A., Morales R.M. Complete dynamic model of a twin rotor MIMO System (TRMS) with experimental validation. In: 39th European Rotorcraft Forum 2013 (ERF 2013); 3–6 Sept. 2013. Moscow, Russia. ISBN: 978-1-5108-1007-5

[14] Ahmad S.M., Chipperfield A.J., Tokhi M.O. Dynamic modelling and open-loop control of a twin rotor multi-input multi-output system. Proceedings of the Institution of Mechanical Engineers Part I, Journal of Systems and Control Engineering. 2002; 216(6): 477–496. doi:10.1177/095965180221600604

[15] Ahmad S.M., Chipperfield A.J., Tokhi M.O. Dynamic modelling and linear quadratic Gaussian control of a twin-rotor multi-input multi-output system. Proceedings of the Institution of Mechanical Engineers Part I, Journal of Systems and Control Engineering. 2003; 217(3):203–227. doi:10.1177/095965180321700304

[16] López-Martínez M., Rubio F.R. Longitudinal control for a laboratory helicopter via constructive approximate backstepping. IFAC Proceedings Volumes. 2005; 38(1):289–294. doi:10.3182/20050703-6-CZ-1902.00448

[17] López-Martínez M., Ortega M.G., Vivas C., Rubio F.R. Nonlinear L_2 control of a laboratory helicopter with variable speed rotors. Automatica. 2007; 43(4): 655–661. doi: 10.1016/j.automatica.2006.10.013

[18] Rahideh A., Bajodah A.H., Shaheed M.H. Real time adaptive nonlinear model inversion control of a twin rotor MIMO system using neural networks. Engineering Applications of Artificial Intelligence. 2012; 25(6):1289–1297. doi:10.1016/j.engappai.2011.12.006

[19] Tao C.W., Taur J.S., Chen Y.C. Design of a parallel distributed fuzzy LQR controller for the twin rotor multi-input multi-output system. Fuzzy Sets and Systems. 2010; 161(15): 2081–2103. doi:10.1016/j.fss.2009.12.007

[20] Reynoso-Meza, G., Garcia-Nieto S., Sanchis J., Blasco, F.X. Controller tuning by means of multi-objective optimization algorithms: a global tuning framework. IEEE Transactions on Control Systems Technology. 2013; 21(2): 445–458. doi:10.1109/TCST.2012.2185698

[21] Coelho J., Matos R., Lebres C., Santos V., Fonseca N.M., Solteiro E.J., Tenreiro J.A. Application of fractional algorithms in the control of a twin rotor multiple input-multiple output system. In: 6th European Nonlinear Dynamics Conference (ENOC 2008). June 30–July 4, 2008. Saint Petersburg, Russia

[22] Christensen R., Fogh N., Hansen R.H., Jensen M.S., Larse S., Paramanathan A. Modelling and control of a twin-rotor MIMO system. Technical report, Aalborg University, Denmark; 2006

[23] Ekbote A.K., Srinivasan N.S., Mahindrakar A.D. Terminal sliding mode control of a twin rotor multiple-input multiple-output system. IFAC Proceedings Volumes. 2011; 44(1):10952–10957. doi:10.3182/20110828-6-IT-1002.00645

[24] Rotondo D., Nejjari F., Puig V. Quasi-LPV modeling, identification and control of a twin rotor MIMO system. Control Engineering Practice. 2013; 21(6): 829–846. doi:10.1016/j.conengprac.2013.02.004

[25] Feedback Co. Twin Rotor MIMO system. Advanced Teaching Manual 1. Manual: 33-007-4M5 Ed01. 1998

[26] Son Y.I., Kim I.H., Choi D.S., Shim D. Robust cascade control of electric motor drives using dual reduced-order PI observer. IEEE Transactions on Industrial Electronics. 2015; 62(6): 3672–3682. doi:10.1109/TIE.2014.2374571

Chapter 12

Synchronization Phenomena in Coupled Birkhoff-Shaw Chaotic Systems Using Nonlinear Controllers

Christos K. Volos, Hector E. Nistazakis,
Ioannis M. Kyprianidis, Ioannis N. Stouboulos and
George S. Tombras

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64811

Abstract

In this chapter, the well-known non-autonomous chaotic system, the Birkhoff-Shaw, which exhibits the structure of beaks and wings, typically observed in chaotic neuronal models, is used in a coupling scheme. The Birkhoff-Shaw system is a second-order non-autonomous dynamical system with rich dynamical behaviour, which has not been sufficiently studied. Furthermore, the master-slave (unidirectional) coupling scheme, which is used, is designed by using the nonlinear controllers to target synchronization states, such as complete synchronization and antisynchronization, with amplification or attenuation in chaotic oscillators. It is the first time that the specific method has been used in coupled non-autonomous chaotic systems. The stability of synchronization is ensured by using Lyapunov function stability theorem in the unidirectional mode of coupling. The simulation results from system's numerical integration confirm the appearance of complete synchronization and antisynchronization phenomena depending on the signs of the parameters of the error functions. Electronic circuitry that models the coupling scheme is also reported to verify its feasibility.

Keywords: chaos, complete synchronization, antisynchronization, anidirectional coupling, nonlinear controller

1. Introduction

In the past decades, the phenomenon of synchronization between coupled nonlinear systems and especially of systems with chaotic behaviour has attracted scientists' interest from all over

the world because it is an interesting phenomenon with a broad range of applications, such as in various complex physical, chemical and biological systems [1–9], in secure and broadband communication system [10, 11] and in cryptography [12, 13].

In synchronization two or more systems with chaotic behaviour can adjust a given of their motion property to a common behaviour (equal trajectories or phase locking), due to forcing or coupling [14]. However, having two chaotic systems being synchronized, it is a major surprise, due to the exponential divergence of the nearby trajectories of the systems. Nevertheless, nowadays the phenomenon of synchronization of coupled chaotic oscillators is well-studied theoretically and proven experimentally.

Synchronization theory has begun studying in the 1980s and early 1990s by Fujisaka and Yamada [15], Pikovsky [16], Pecora and Carroll [17]. Onwards, a great number of research works based on synchronization of nonlinear systems has risen and many synchronization schemes depending on the nature of the coupling schemes and of the interacting systems have been presented. Complete or full chaotic synchronization [18–23], phase synchronization [24, 25], lag synchronization [26, 27], generalized synchronization [28], antisynchronization [29, 30], anti-phase synchronization [31–36], projective synchronization [37], anticipating [38] and inverse lag synchronization [39] are the most interesting types of synchronization, which have been investigated numerically and experimentally by many research groups.

This chapter deals with two of the aforementioned cases: the complete synchronization and the antisynchronization. In the case of complete synchronization, two identically coupled chaotic systems have a perfect coincidence of their chaotic trajectories, i.e., $x_1(t) = x_2(t)$ as $t \to \infty$. In the case of antisynchronization, for initial conditions chosen from large regions in the phase space two coupled systems x_1 and x_2, can be synchronized in amplitude, but with opposite sign, that is $x_1(t) = -x_2(t)$ as $t \to \infty$.

From our knowledge, chaotic systems exhibit high sensitivity on initial conditions or system's parameters and if they are identical and start from almost the same initial conditions, they follow trajectories which rapidly become uncorrelated. That is why many techniques exist to obtain chaotic synchronization. So, many of these techniques for coupling two or more nonlinear chaotic systems can be mainly divided into two classes: unidirectional coupling and bidirectional or mutual coupling [40]. In the first case, only the first system, the master system, drives the second one, the slave system, while in the second case, each system's dynamic behaviour influences the dynamics of the other.

Furthermore, the subject of synchronization between coupled chaotic systems, especially in the last decade, plays a crucial role in the field of neuronal dynamics [6, 41]. Neural signals in the brain are observed to be chaotic and it is worth considering further their possible synchronization [42–46]. These signals are produced by nerve membranes exhibiting their own nonlinear dynamics, which generate and propagate action potentials. Such nonlinear dynamics in nerve membranes can produce chaos in neurons and related bifurcations.

So, motivated by the aforementioned fact, the Birkhoff-Shaw system [45], which exhibits the structure of beaks and wings, typically observed in chaotic neuronal models, is chosen for use in this chapter. It is a second order non-autonomous dynamical system with rich dynamical

http://dx.doi.org/10.5772/64811

behaviour, which has not been sufficiently studied. Furthermore, the unidirectional coupling scheme, which is used, is designed by using the nonlinear controllers to target synchronization states, such as complete synchronization and antisynchronization, with amplification or attenuation in chaotic oscillators. The stability of synchronization is ensured by using Lyapunov function stability theorem in the unidirectional mode of coupling. The simulation results from system's numerical integration confirm the appearance of complete synchronization and antisynchronization phenomena depending on the signs of the parameters of the error functions. Electronic circuitry that models the coupling scheme is also reported to verify its feasibility.

This chapter is organized as follows. In Section 2, the features of chaotic systems and especially of the proposed Birkhoff-Shaw system by using various tools of nonlinear theory, such as bifurcation diagrams, phase portraits and Lyapunov exponents, are explored. The synchronization scheme, by using the nonlinear controller, as well as the unidirectional coupling scheme is discussed in Sections 3 and 4, respectively. The simulation results of the proposed method are presented for various cases in Section 5. Section 6 presents the circuital implementation of the coupling scheme and the results which are obtained by using the SPICE. Finally, the conclusive remarks and some thoughts for future works are drawn in the last section.

2. The Birkhoff-Shaw chaotic system

As it is known, chaos theory studies systems that present three very important features [46, 47]:

- its periodic orbits must be dense,
- it must be topologically mixing and
- it must be very sensitive on initial conditions.

In more details, the periodic orbits of a chaotic system have to be dense and that means that the trajectory of a dynamical system is dense, if it comes arbitrarily close to any point in the domain. The second feature of chaotic systems, the topological mixing, means that the chaotic trajectory at the phase space will move over time so that each designated area of this trajectory will eventually cover part of any particular region. Additionally, the third feature, which is the most important feature of chaotic systems, is the sensitivity on initial conditions. When a small variation on a system's initial conditions exists, a totally different chaotic trajectory will be produced.

Here, as it is mentioned above, the well-known non-autonomous chaotic system of Birkhoff-Shaw, which has been proposed by Shaw in 1981 [45], is used. The Birkhoff-Shaw system is described by the 2-D system of differential equations:

$$\begin{cases} \dot{x} = ay + x - cxy^2 \\ \dot{y} = -x - B\cos(dt) \end{cases} \tag{1}$$

where x and y are the states variables and a, B, c and d are positive parameters.

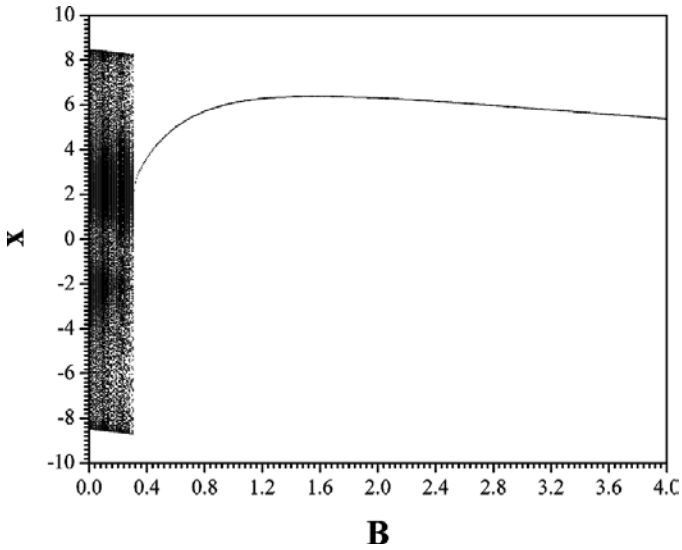

Figure 1. Bifurcation diagram of x versus B, for $a = 1$, $c = 0.1$ and $d = 1$.

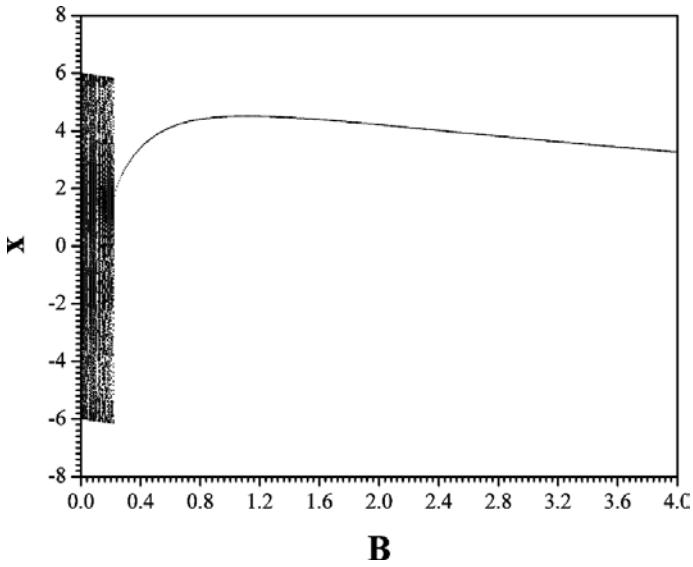

Figure 2. Bifurcation diagram of x versus B, for $a = 1$, $c = 0.2$ and $d = 1$.

In this section, the system's dynamic behaviour is investigated numerically by employing a fourth order Runge-Kutta algorithm. As a first step in this approach, the bifurcation diagram and the Lyapunov exponents, which are very useful tools from nonlinear theory, are used. In

http://dx.doi.org/10.5772/64811

Figures 1–8, two sets of bifurcation diagrams of the variable x versus the parameter B, for c = 0.1 and c = 0.2 and for various values of the parameter d, are displayed. The above bifurcation diagrams show the richness of system's dynamical behaviour. Apart from limit cycles, system (1) has quasiperiodicity and chaos, which makes the system's control a difficult target in practical applications where a particular dynamic is desired.

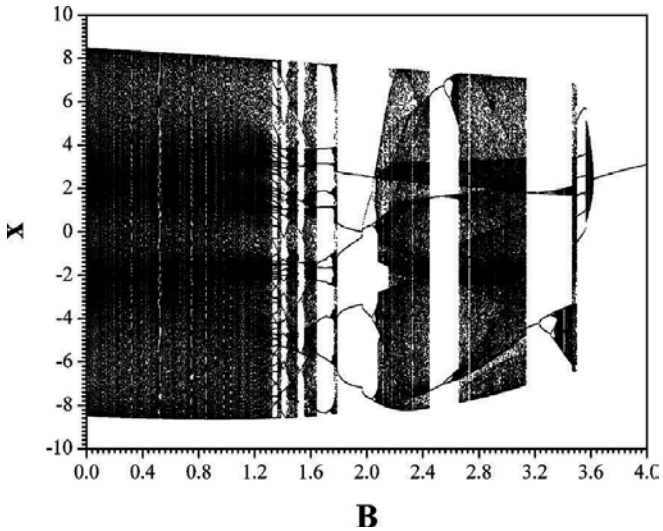

Figure 3. Bifurcation diagram of x versus B, for a = 1, c = 0.1 and d = 1.5.

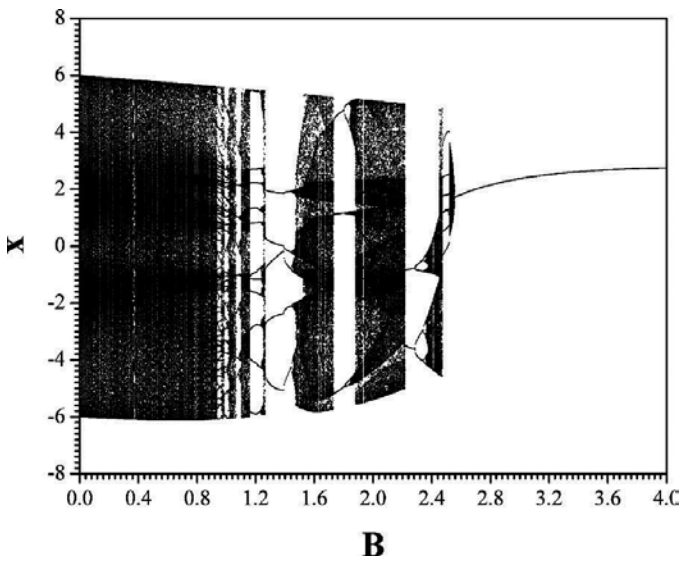

Figure 4. Bifurcation diagram of x versus B, for a = 1, c = 0.2 and d = 1.5.

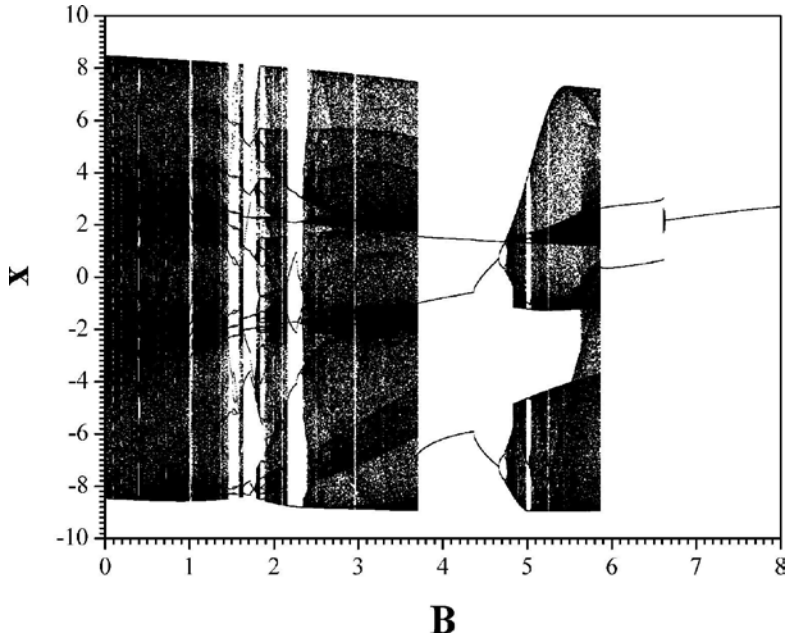

Figure 5. Bifurcation diagram of x versus B, for $a = 1$, $c = 0.1$ and $d = 2$.

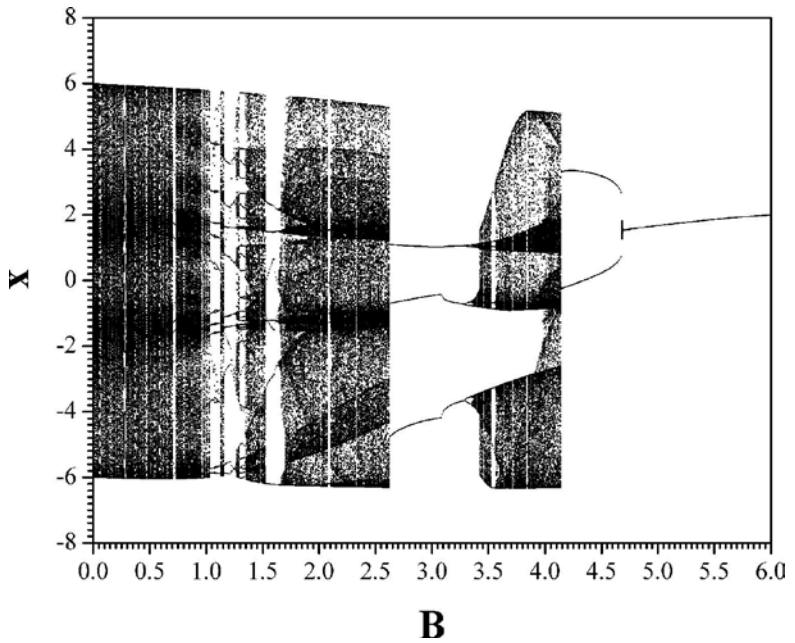

Figure 6. Bifurcation diagram of x versus B, for $a = 1$, $c = 0.2$ and $d = 2$.

http://dx.doi.org/10.5772/64811

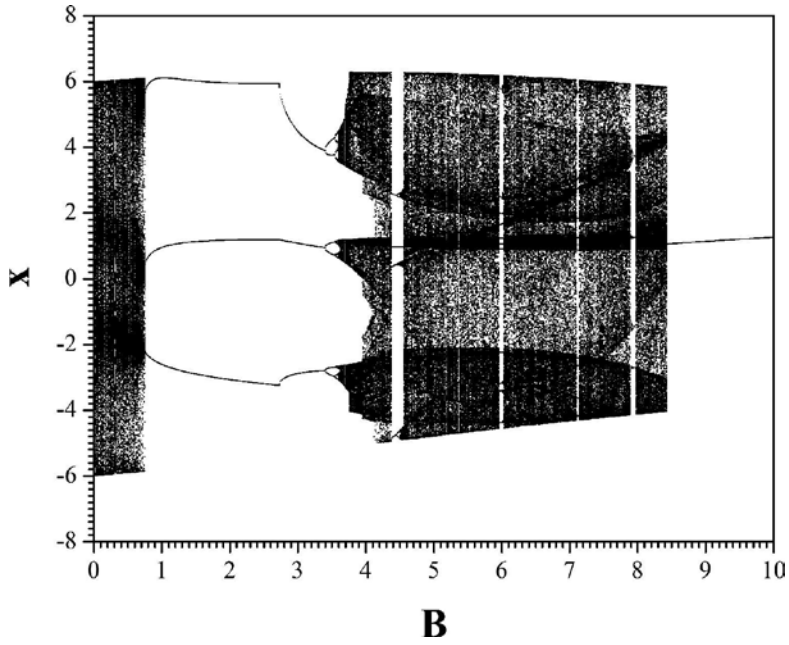

Figure 7. Bifurcation diagram of x versus B, for $a = 1$, $c = 0.1$ and $d = 3$.

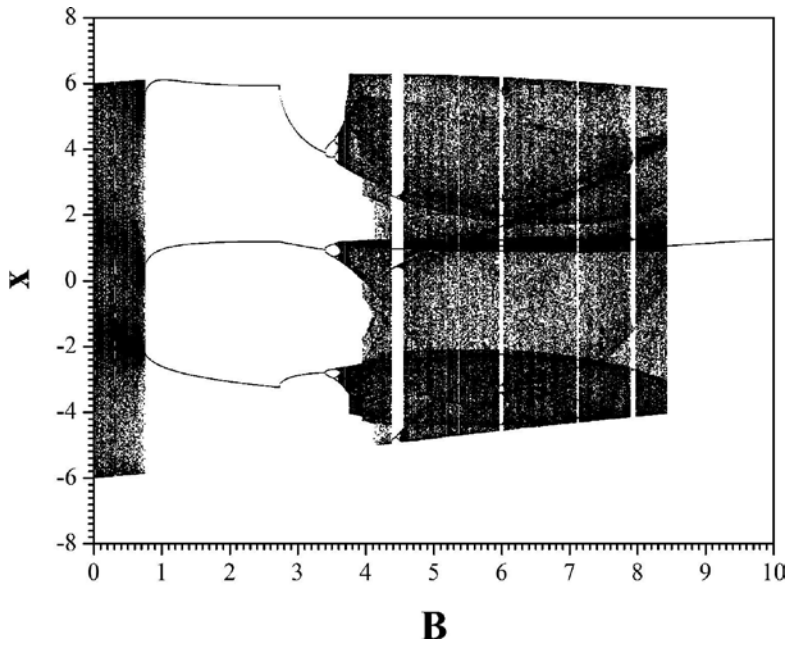

Figure 8. Bifurcation diagram of x versus B, for $a = 1$, $c = 0.2$ and $d = 3$.

In greater detail, having small values of the parameter d (i.e. $d = 1$) the system begins from a quasiperiodic state and as the amplitude B of the external force increases, the system passes to a stable periodic behaviour of period-1 (**Figures 1** and **2**). For example, in the case of $a = 1$, $c = 0.1$ and $d = 1$, the Lyapunov exponents (LEs) for two respective values of B in the regions of quasiperiodic and periodic regions are:

- for $B = 0.1$ (quasiperiodic state): $LE_1 = 0.000$, $LE_2 = 0.000$, $LE_3 = -1.516$
- for $B = 2$ (periodic state): $LE_1 = 0.000$, $LE_2 = -0.996$, $LE_3 = -80.998$

According to the nonlinear theory, if the number of zeros of LEs is one or two then the system is in periodic or quasiperiodic behaviour, respectively. So, the calculation of Lyapunov exponents plays a crucial role to the estimation of the dynamic behaviour of the proposed system.

However, as the value of the parameter d increases the system's complexity is also increased. For $d = 1.5$ (**Figures 3** and **4**) in both cases of $c = 0.1$ and $c = 0.2$, the range of quasiperiodic region has been significantly enlarged, as compared to the previous case ($d = 1$). Nevertheless, with the end of this region, system's behaviour alternates between periodic and chaotic ones. The chaotic regions are detected by finding one positive Lypaunov exponent (i.e. for $a = 1$, $B = 2.8$, $c = 0.1$ and $d = 1.5$, the Lypaunov exponents are: $LE_1 = 0.157$, $LE_2 = 0.000$, $LE_3 = -1.626$). Finally, the system passes from a quasiperiodic state to a stable periodic (period-1) one again.

System's behaviour remains almost the same as the value of parameter d (i.e. $d = 2$) increases (**Figures 5** and **6**). However, two important conclusions could be drawn. The first is that the chaotic regions have been enlarged, while the second is that the quasiperiodic region, before the final system's periodic state, has been significantly decreased.

Finally, if the value of parameter d has been further increased(i.e. $d = 3$) then the chaotic regions have also been increased while the respective periodic regions have been significantly decreased. Also, the system suddenly passes from chaotic to the final periodic behaviour, as it is shown in the bifurcation diagram of **Figures 7** and **8**.

In these diagrams, the region of period-3 dominates, which is characteristic of system's chaotic behaviour. Also, this region reveals two more important phenomena from nonlinear theory. Firstly, this window of period-3 begins with a sudden transition from a chaotic to periodic behaviour, which in this case is known as *Intermittency* [48] and ends with an *Interior Crisis* [49, 50] that causes intermittency induced from crisis.

In **Figures 9–12**, the phase portraits for various values of the parameter B, in the case of $a = 1$, $c = 0.2$ and $d = 3$, are presented. In more details, **Figure 9** shows the quasiperiodic attractor, that the system is in for low values of the amplitude B ($B = 0.5$) of the external sinusoidal source, while **Figures 10** and **12** display the system's periodic attractors of period-3 ($B = 3$) and period-1 ($B = 9$), respectively. Finally, in **Figure 11** the system's chaotic attractor for $B = 7$ is presented.

http://dx.doi.org/10.5772/64811

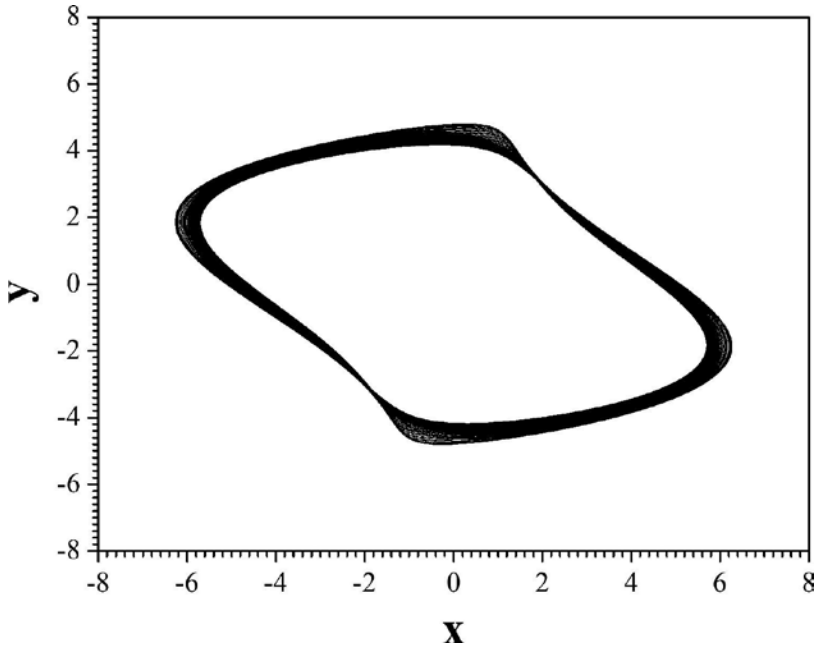

Figure 9. Phase portrait of y versus x, for $a = 1$, $c = 0.2$, $d = 3$ and $B = 0.5$ (quasiperiodic behaviour).

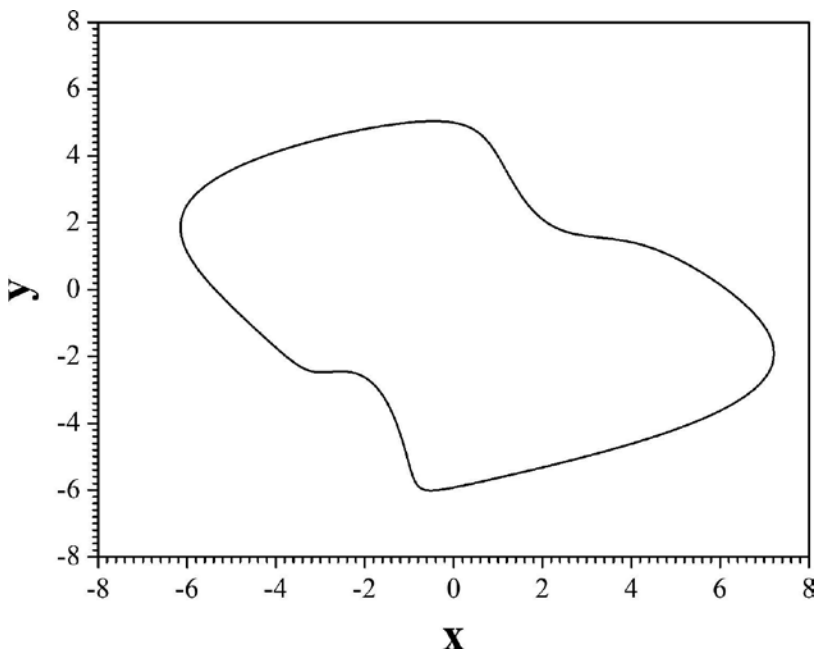

Figure 10. Phase portrait of y versus x, for $a = 1$, $c = 0.2$, $d = 3$ and $B = 3$ (periodic behaviour).

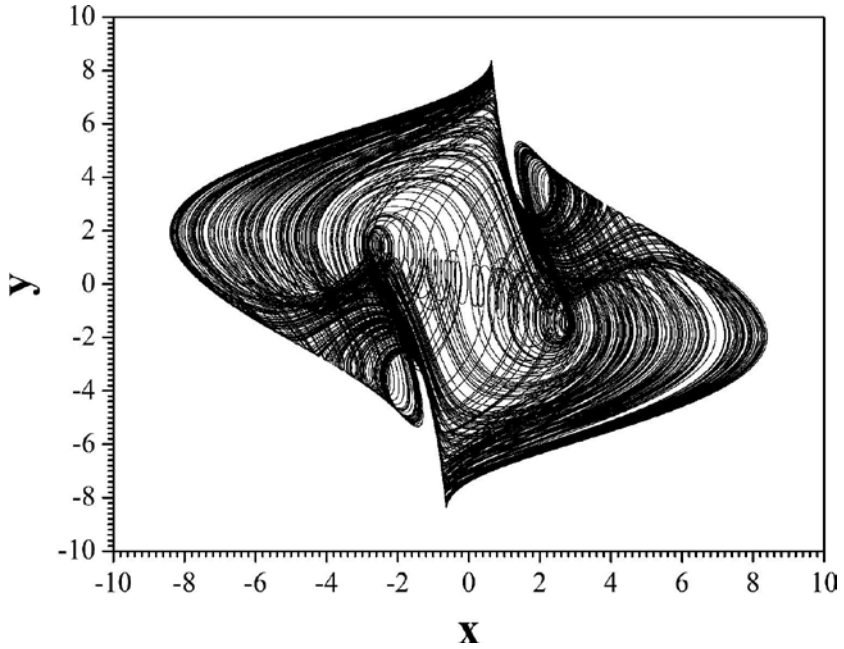

Figure 11. Phase portrait of y versus x, for $a = 1$, $c = 0.2$, $d = 3$ and $B = 7$ (chaotic behaviour).

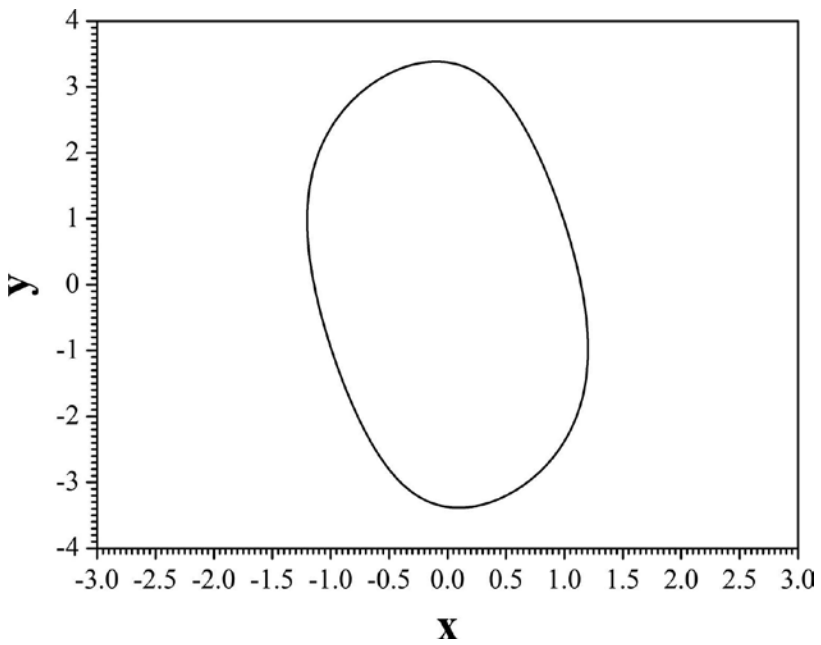

Figure 12. Phase portrait of y versus x, for $a = 1$, $c = 0.2$, $d = 3$ and $B = 9$ (periodic behaviour).

http://dx.doi.org/10.5772/64811

3. The proposed coupling scheme

Two identical unidirectionally coupled chaotic systems can be described by the following system of differential equations:

$$\begin{cases} \dot{x} = f(x) + U_X \\ \dot{y} = f(y) + U_Y \end{cases} \tag{2}$$

where $(f(x), f(y)) \in R^n$ are the flows of the systems. Nonlinear controllers (NCs), U_X and U_Y, define the coupling of the systems, while the error function is given by $e = ky - lx$, where k and l are constants [51, 52]. If the Lyapunov function stability (LFS) technique is applied, a stable synchronization state will be obtained when the error function of the coupled system follows the limit:

$$\lim_{t\to\infty} \|e(t)\| \to 0 \tag{3}$$

so that $lx = ky$.

The design process of the coupling scheme, is based on the Lyapunov function:

$$V(e) = \frac{1}{2} e^T e \tag{4}$$

where T is a transpose of a matrix and $V(e)$. The Lyapunov function (4) is a positive definite function. Also, for known system's parameters and with the appropriate choice of the controllers U_X and U_Y, the coupled system has $V(e) < 0$. This ensures the asymptotic global stability of synchronization and thereby realizes any desired synchronization state [51, 52].

By using the appropriate NCs functions U_X, U_Y and error function's parameters k, l, a bidirectional (mutual) or unidirectional coupling scheme can be implemented. Analytically, while if $U_{X,Y} \neq 0$ and $k, l \neq 0$, a bidirectional coupling scheme is realized, while if ($U_X = 0$, $k = 1$) or ($U_Y = 0$, $l = 1$), a unidirectional coupling scheme is realized, respectively. The signs of the constants k, l play a crucial role to the synchronization case (complete synchronization or antisynchronization), which is observed in this work. However, the ratio of k over l decides the amplification of one oscillator relative to another one.

Next, the simulation results in the unidirectional coupling scheme and for various values of parameters k and l are presented in details.

4. Unidirectional coupling

In this section, the unidirectional coupling scheme for $U_X = 0$, in the case of coupled systems of Eq. (1), is presented. The coupled system is described by the following systems of Eqs. (5) and (6).

Master system:

$$\begin{cases} \dot{x}_1 = ax_2 + x_1 - cx_1x_2^2 \\ \dot{x}_2 = -x_1 - B\cos(dt) \end{cases} \tag{5}$$

Slave system:

$$\begin{cases} \dot{y}_1 = ay_2 + y_1 - cy_1y_2^2 + U_{Y1} \\ \dot{y}_2 = -y_1 - B\cos(dt) + U_{Y2} \end{cases} \tag{6}$$

where $\boldsymbol{U}_Y = [U_{Y1}, U_{Y2}]^T$ is the Nonlinear Controller (NC). The error function is defined by $\boldsymbol{e} = k\boldsymbol{y} - l\boldsymbol{x}$, with $\boldsymbol{e} = [e_1, e_2]^T$, $\boldsymbol{x} = [x_1, x_2]^T$ and $\boldsymbol{y} = [y_1, y_2]^T$. So, the error dynamics, by taking the difference of Eqs. (5) and (6), are written as:

$$\begin{cases} \dot{e}_1 = ae_2 + e_1 + lcx_1x_2^2 - kcy_1y_2^2 + kU_{Y1} \\ \dot{e}_2 = -e_1 - B(k-l)\cos(dt) + kU_{Y2} \end{cases} \tag{7}$$

For stable synchronization, $e \to 0$ as $t \to \infty$. By substituting the conditions in Eq. (7) and taking the time derivative of Lyapunov function

$$\begin{aligned} &\dot{V}(e) = e_1\dot{e}_1 + e_2\dot{e}_2 = \\ &= e_1\left(ae_2 + e_1 + lcx_1x_2^2 - kcy_1y_2^2 + kU_{Y1}\right) + e_2\left(-e_1 - B(k-l)cos(dt) + kU_{Y2}\right) \end{aligned} \tag{8}$$

We consider the following NC controllers:

$$\begin{cases} U_{Y1} = -\dfrac{1}{k}(ae_2 + 2e_1 + lcx_1x_2^2 - kcy_1y_2^2) \\ U_{Y2} = -\dfrac{1}{k}\left(-e_1 - B(k-l)\cos(dt) + e_2\right) \end{cases} \tag{9}$$

such that

http://dx.doi.org/10.5772/64811

$$\dot{V}(e) = -e_1^2 - e_2^2 < 0 \tag{10}$$

Eq. (10) ensures the asymptotic global stability of synchronization.

5. Simulation results

In this section, the simulation results, with the unidirectional coupling scheme, in three different cases are presented.

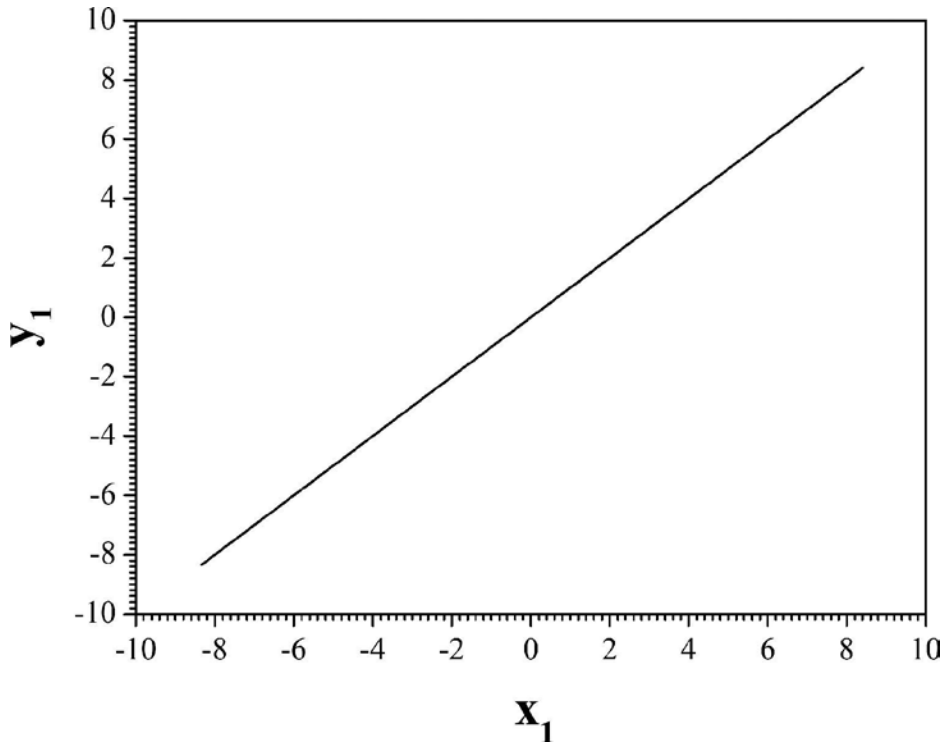

Figure 13. The phase portrait of y_1 versus x_1, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

5.1. The case for $k = l = 1$

As it is mentioned, the phenomenon of complete synchronization is achieved for every value of k, l. Especially for $k = -l = 1$, the two coupled systems are in the chaotic state, due to the chosen values of system's parameters ($a = 1$, $B = 7$, $c = 0.2$ and $d = 3$) and initial conditions (x_1, x_2, y_1, y_2) = (3, 2, –1, –5). The goal of complete synchronization is achieved as it is shown from the plots of y_1 versus x_1, the time-series of x_2, y_2 and the errors e_i in **Figures 13–15**.

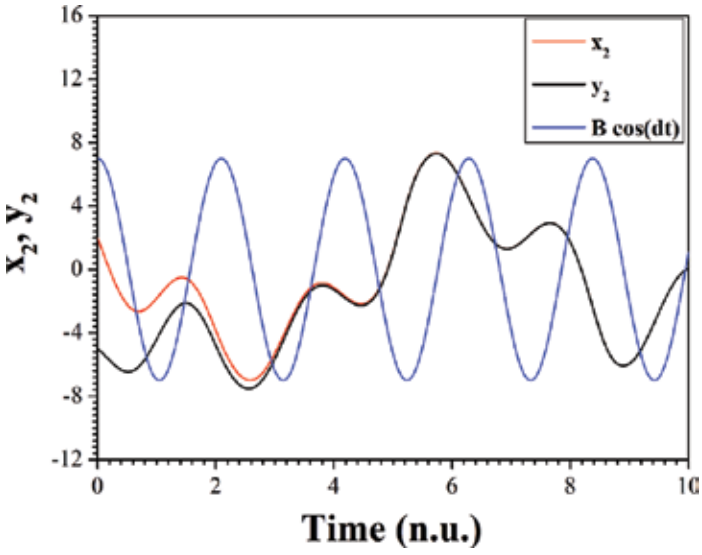

Figure 14. The time-series of x_2, y_2, in regards to the external periodic signal, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

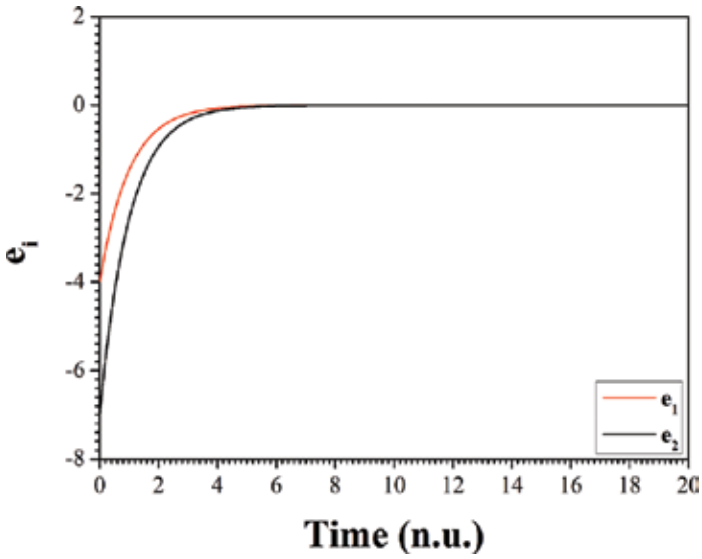

Figure 15. The time-series of errors e_1, e_2, with $k = l = 1$, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

5.2. The case for $k = l = 1$

In the second case, by using opposing values for the parameters $k = -l = 1$ and for the same values of system's parameters ($a = 1$, $B = 7$, $c = 0.2$ and $d = 3$), the phenomenon of antisynchronization is achieved. This conclusion is derived from the phase portrait of y_1 versus x_1

http://dx.doi.org/10.5772/64811

(**Figure 16**), as well as from the time series of x_2, y_2 (**Figure 17**). Also, the plot of errors $e_i = y_i + x_i$ in **Figure 18** confirms the antisynchronization of the coupled system.

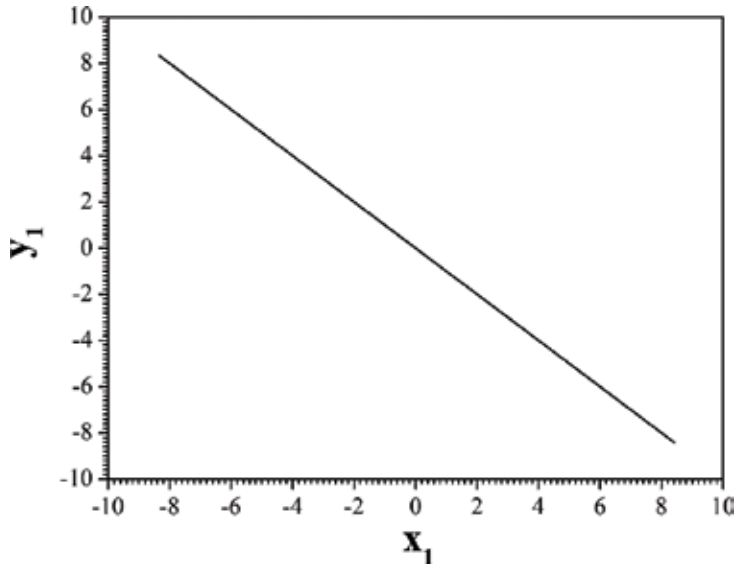

Figure 16. The phase portrait of y_1 versus x_1, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

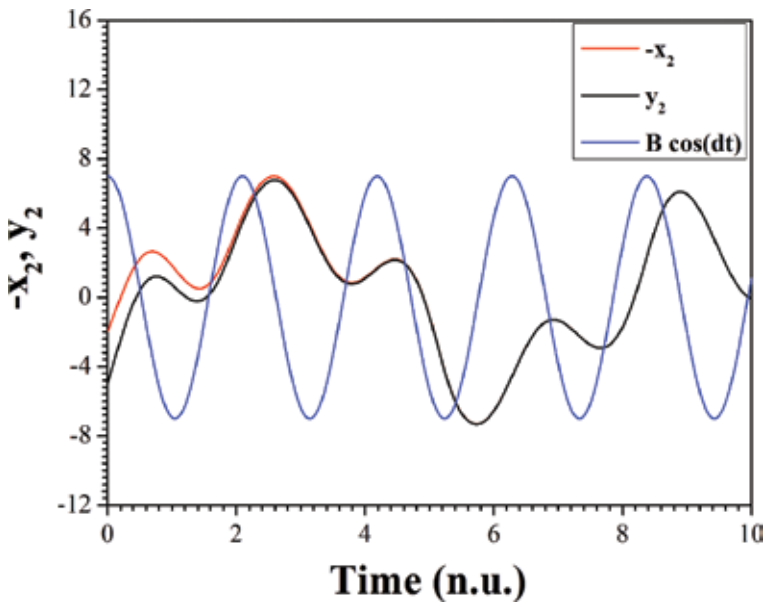

Figure 17. The time-series of $-x_2$, y_2, in regard to the external periodic signal, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

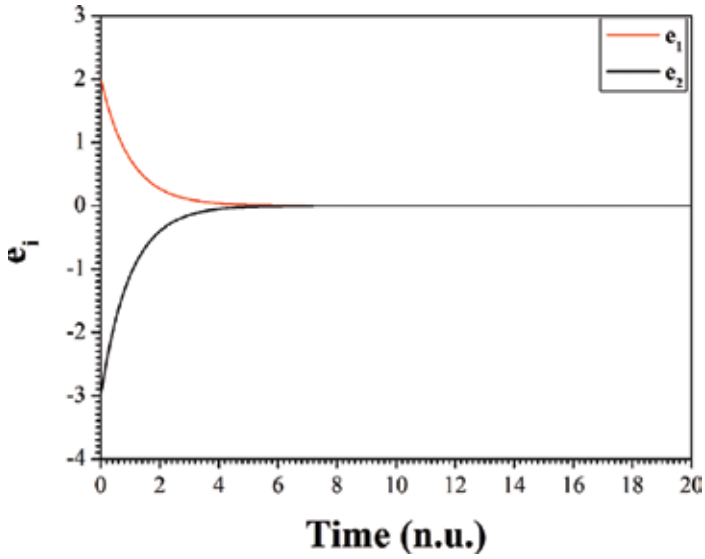

Figure 18. The time-series of errors e_1, e_2, with $k = l = 1$, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

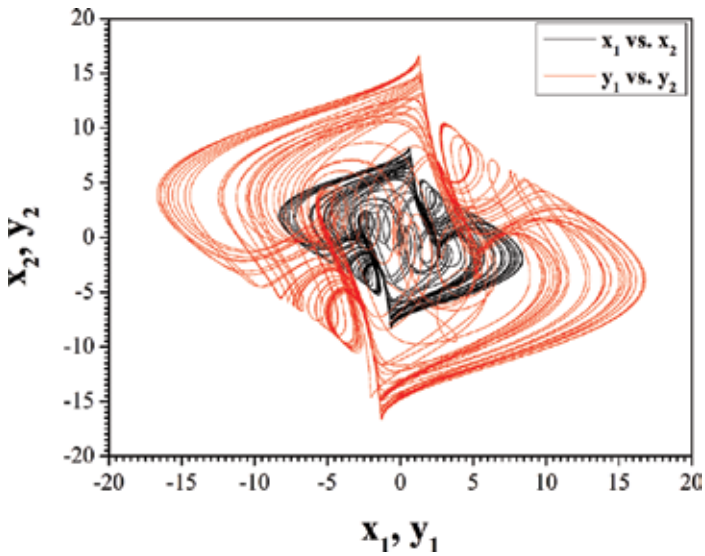

Figure 19. The phase portraits of x_2 versus x_1(black colour) and y_2 versus y_1 (red colour), for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

5.3. The case for $k = 1$, $l = 2$

In this case, the parameters of the error functions are chosen as $k = 1$ and $l = 2$. By choosing the systems' parameters as $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$ the chaotic attractor of the second system is enlarged by two times, as it is shown with red colour in **Figure 19**, as well as by the time-

http://dx.doi.org/10.5772/64811

series of signals x_2 and y_2 (**Figure 21**). The y_1 versus x_1 plot in **Figure 20** confirms that the coupled system is in complete synchronization state independently of the values of the error's parameters k, l. The error plot $e_i = y_i - 2x_i$ ($i = 1, 2$) in **Figure 22** shows the exponential convergence to zero that confirms the realization of system's complete synchronization state.

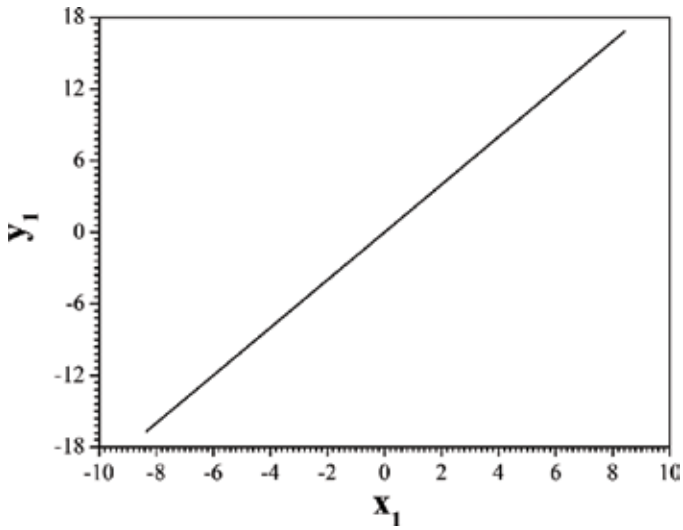

Figure 20. The phase portrait of y_1 versus x_1, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

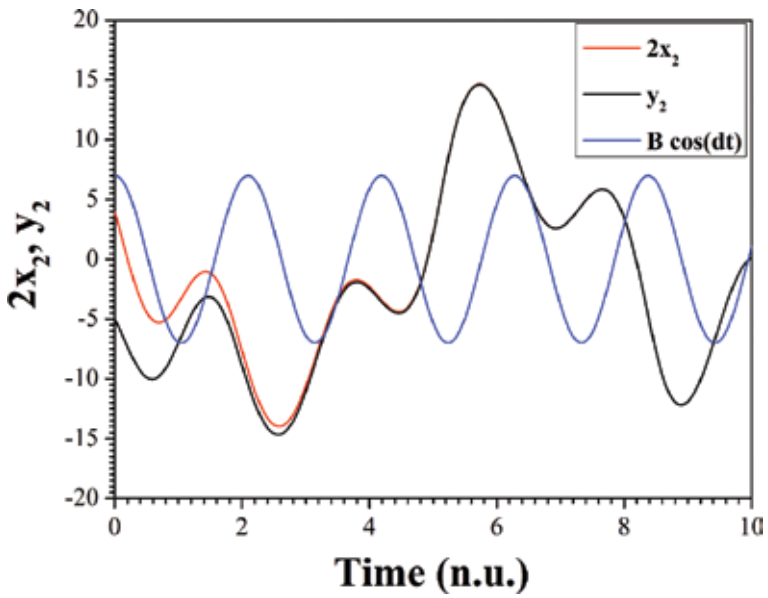

Figure 21. The time-series of $2x_2$, y_2, in regard to the external periodic signal, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

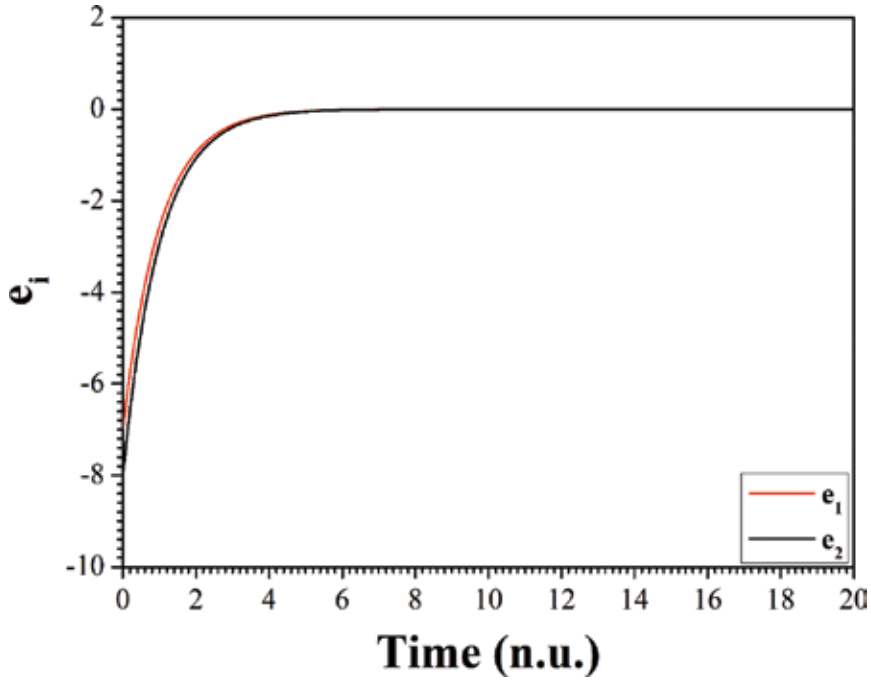

Figure 22. The time-series of errors e_1, e_2, with $k = 1$, $l = 2$, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$.

6. Circuit's implementation of the coupling scheme

The circuit implementation of the proposed synchronization coupling scheme, with the electronic simulation package Cadense OrCAD, for $k = l = 1$, is presented in this section, in order to prove the feasibility of the proposed method. The coupling system's circuitry design consists of three sub-circuits, which are the master circuit, the coupling circuit and the slave circuit. Also, the circuit is realized by using common electronic components.

Figure 23 shows the schematic of the master circuit, which has two integrators (U_1 and U_2) and one differential amplifier (U_3), which are implemented with the TL084, as well as two signals multipliers (U_4, U_5) by using the AD633. By applying Kirchhoff's circuit laws, the corresponding circuital equations of designed master circuit can be written as:

$$\begin{cases} \dot{x}_1 = \dfrac{1}{RC}\left(x_2 + x_1 - \dfrac{R}{100R_1} x_1 x_2^2 \right) \\ \dot{x}_2 = \dfrac{1}{RC}\left(-x_1 - V_0 \cos(\omega t) \right) \end{cases} \tag{11}$$

http://dx.doi.org/10.5772/64811

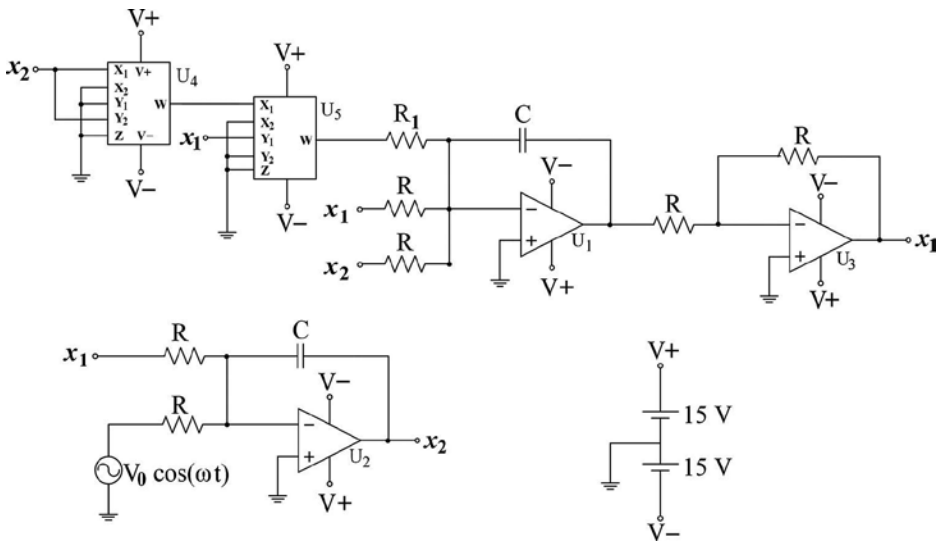

Figure 23. The schematic representation of the master circuit.

where x_i (i = 1, 2) are the voltages in the outputs of the operational amplifiers U_3 and U_2. Normalizing the differential equations of system (18) by using $\tau = T/RC$ we could see that this system is equivalent to the system (12). The circuit components have been selected as: R = 10 kΩ, R_1 = 500 Ω, C = 10 nF, V_0 = 7 V and f = 4777 Hz, while the power supplies of all active devices are ±17 V_{DC}. For the chosen set of components the master system's parameters are: a = 1, B = 7, c = 0.2 and d = 3. In **Figure 24**, the chaotic attractor, which is obtained from Cadence OrCAD in (x_1, x_2) phase plane, is proved to be in a very good agreement with the respective phase portrait from system's numerical simulation process (**Figure 11**). So, the proposed circuit emulates very well the master system.

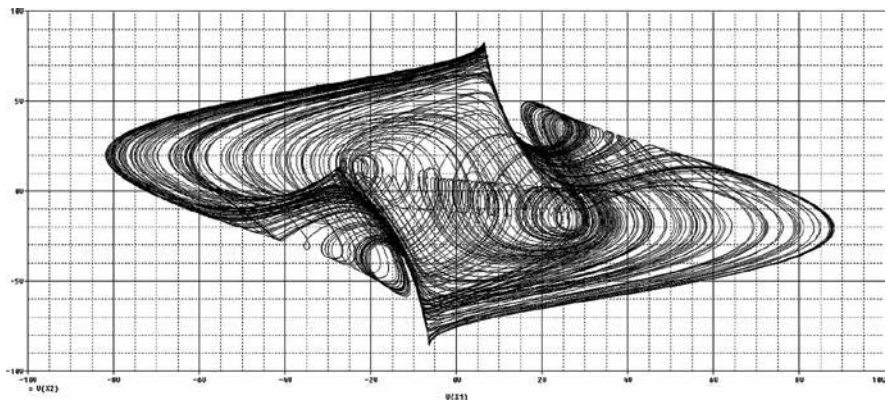

Figure 24. The chaotic attractor produced by the designed master circuit, obtained from Cadence OrCAD in the (x_1, x_2) phase plane.

In **Figure 25**, the schematic of the slave circuit, which is similar to the master circuit, is shown. The difference of this circuit in comparison to the previous one are the signals u_1 and mu_2, where u_1 is the control signal U_{Y1} and mu_2 is the opposite, due to the integrator, of the signal U_{Y2}, of system (6). So, for $k = l = 1$, the signal mu_2 is given as

$$mu_2 = -e_1 + e_2 \tag{12}$$

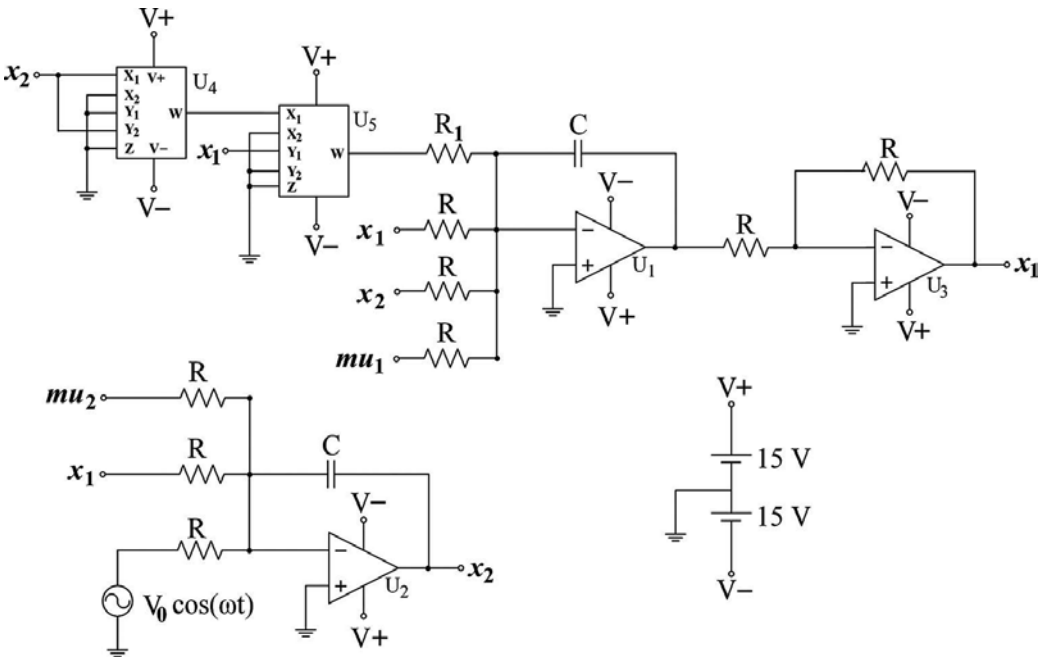

Figure 25. The schematic representation of the slave circuit.

The dynamics of the slave circuit is described by the following set of differential equations.

$$\begin{cases} \dot{y}_1 = \dfrac{1}{RC}\left(y_2 + y_1 - \dfrac{R}{100R_1} y_1 y_2^2 + u_1 \right) \\ \dot{y}_2 = \dfrac{1}{RC}\left(-y_1 - V_0 \cos(\omega t) - mu_2 \right) \end{cases} \tag{13}$$

Finally, the units from which the coupling circuit is consisted, are shown in the schematic of **Figure 26**, in which e_i, ($i = 1, 2$) are the difference signals ($e_i = ky_i - lx_i$, $i = 1, 2$), with $k = l = 1$ and me_2 is the opposite of e_2. Also, the resistors $R_2 = 5\,\text{k}\Omega$ and $R_3 = 50\,\text{k}\Omega$ have been used for achieving the desired values of system's parameters.

http://dx.doi.org/10.5772/64811

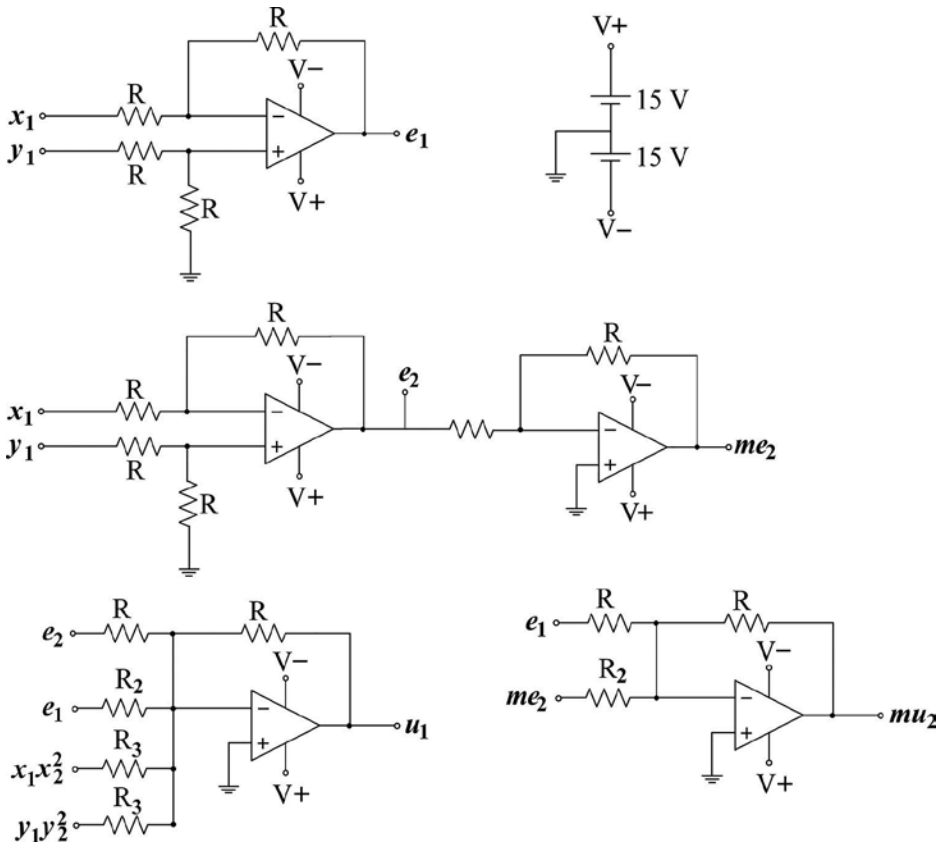

Figure 26. The schematic representation of the coupling circuit.

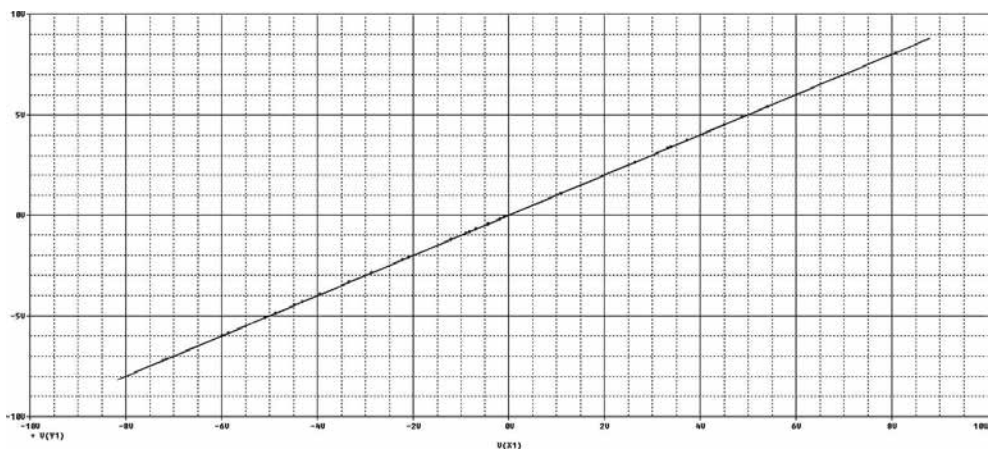

Figure 27. The phase portrait of y_1 vs. x_1, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$, obtained from Cadence OrCAD.

Figures 27 and **28** depict the phase portraits in (x_i, y_i) phase planes, with $i = 1, 2$, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$, obtained from Cadence OrCAD. These figures confirm the achievement of complete synchronization in the case of unidirectionally coupled circuits with the proposed method.

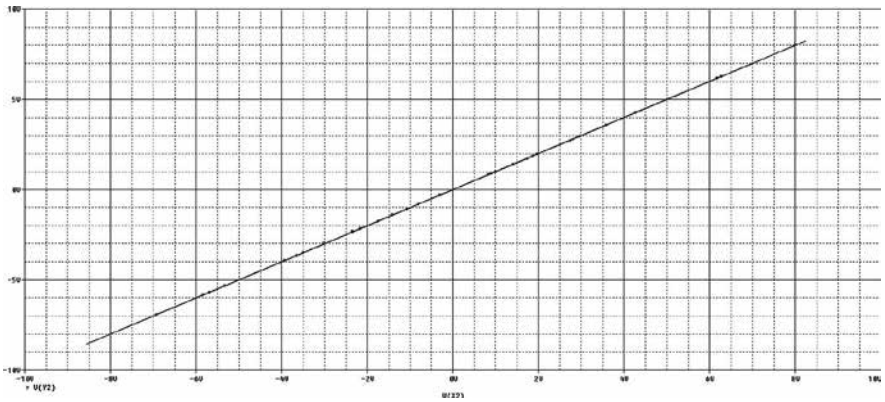

Figure 28. The phase portrait of y_2 versus x_2, for $a = 1$, $B = 7$, $c = 0.2$ and $d = 3$, obtained from Cadence OrCAD.

7. Conclusion

In this chapter, the case of unidirectional coupling scheme of two chaotic non-autonomous dynamical systems was studied. The proposed system is the second order Birkhoff-Shaw system, which is simple but very interesting from the perspective of nonlinear analysis. Furthermore, the coupling method was based on a recently new proposed scheme based on the nonlinear controller, which is applied for the first time in non-autonomous systems.

The Birkhoff-Shaw system is one of the simplest 2-D nonlinear systems exhibiting a rich dynamical behaviour. Besides limit cycles, Birkhoff-Shaw system presents quasiperiodicity and chaos, which can make the control of the system a difficult target in practical applications, where a particular dynamic is desired. Also, two well-known phenomena of nonlinear theory, the Intermittency and the Interior Crisis have been observed. However, the main drawback of this system is the fact that this system is a non-autonomous dynamical system, which makes the coupling method weak, especially if it is used in secure communication schemes.

In agreement to the simulation results, the circuital implementation of the proposed system in SPICE, in the case of unidirectional coupling, confirms the appearance of complete synchronization and antisynchronization, depending on the signs of the parameters of the error functions, in various cases. With this method, by choosing an appropriate sign for the error functions, the coupling system can be driven either to complete synchronization or antisynchronization behaviour.

http://dx.doi.org/10.5772/64811

From our knowledge, the complex behaviour of chaotic systems, like the ones that mentioned above, makes the synchronization difficult in practical applications where a particular dynamic is desired. For this reason, the synchronization of chaotic systems has attracted considerable attention due to its great potential applications, in secure communication, chemical reactions and biological systems. Especially, the synchronization in coupled neurons is a subject of a growing interest in the research community. So, due to the fact that Birkhoff-Shaw chaotic attractor exhibits the structure of beaks and wings, typically observed in chaotic neuronal models, the proposed coupling scheme showed an interesting research result of achieving the synchronization or antisynchronization in the case of coupled neuronal models.

As a next step in this direction is the application of the proposed method in non-identical Birkhoff-Shaw coupled systems in order to satisfy the goal of control of systems, which are in totally different dynamical behaviours. Also, the case of bidirectional coupling as well as the case of generalized synchronization, with the proposed scheme, could be examined.

Author details

Christos K. Volos[1], Hector E. Nistazakis[2*], Ioannis M. Kyprianidis[1], Ioannis N. Stouboulos[1] and George S. Tombras[2]

*Address all correspondence to: enistaz@phys.uoa.gr

1 Physics Department, Aristotle University of Thessaloniki, Thessaloniki, Greece

2 Department of Electronics, Computers, Telecommunications and Control, Faculty of Physics, National and Kapodistrian University of Athens, Athens, Greece

References

[1] Holstein-Rathlou NH, Yip KP, Sosnovtseva OV, Mosekilde E. Synchronization phenomena in nephron-nephron interaction. Chaos. 2001;11:417–426.

[2] Mosekilde E, Maistrenko Y, Postnov D, editors. Chaotic synchronization: applications to living systems. Singapore: World Scientific; 2002. 440 p.

[3] Pikovsky AS, Rosenblum M, Kurths J, editors. Synchronization: a universal concept in nonlinear sciences. Cambridge: Cambridge University Press; 2003. 433 p.

[4] Szatmári I, Chua LO. Awakening dynamics via passive coupling and synchronization mechanism in oscillatory cellular neural/nonlinear networks. Int. J. Circ. Theor. Appl. 2008;36:525–553.

[5] Tognoli E, Kelso JAS. Brain coordination dynamics: true and false faces of phase synchrony and metastability. Prog. Neurobiol. 2009;87:31–40.

[6] Wang J, Che YQ, Zhou SS, Deng B. Unidirectional synchronization of Hodgkin-Huxley neurons exposed to ELF electric field. Chaos Solit. Fract. 2009;39:1335–1345.

[7] Gerodimos NA, Daltzis PA, Hanias MP, Nistazakis HE, Tombras GS. Unimodal 1-D maps cousins in nature. New research trends in nonlinear circuits: design, chaotic phenomena and applications. ISBN: 978–1–3321–406–4, New York: Nova Publishers; 2014.

[8] Liu X, Chen T. Synchronization of identical neural networks and other systems with an adaptive coupling strength. Int. J. Circ. Theor. Appl. 2010;38:631–648.

[9] Hanias MP, Nistazakis HE, Tombras GS. Optoelectronic chaotic circuits. Optoelectronic devices and properties. Croatia: Intech Publishers; 2013. ISBN: 978–953–307–204–3.

[10] Jafari S, Haeri M, Tavazoei MS. Experimental study of a chaos-based communication system in the presence of unknown transmission delay. Int. J. Circ. Theor. Appl. 2010;38:1013–1025.

[11] Dimitriev AS, Kletsovi AV, Laktushkin AM, Panas AI, Starkov SO. Ultrawideband wireless communications based on dynamic chaos. J. Commun. Technol. Electron. 2006;51:1126–1140.

[12] Grassi G, Mascolo S. Synchronization of high-order oscillators by observer design with application to hyperchaos-based cryptography. Int. J. Circ. Theor. Appl. 1999;27:543–553.

[13] Volos CK, Kyprianidis, IM, Stouboulos IN. Experimental demonstration of a chaotic cryptographic scheme. WSEAS Trans. Circ. Syst. 2006;5:1654–1661.

[14] Luo ACJ, editor. Dynamical system synchronization. New York: Springer; 2013. 239 p.

[15] Fujisaka H, Yamada T. Stability theory of synchronized motion in coupled-oscillator systems. Prog. Theor. Phys. 1983;69:32–47.

[16] Pikovsky AS. On the interaction of strange attractors. Z. Phys. B: Condensed Matter. 1984;55:149–154.

[17] Pecora LM, Carroll TL. Synchronization in chaotic systems. Phys. Rev. Lett. 1990;64:521–524.

[18] Maritan A, Banavar J. Chaos noise and synchronization. Phys. Rev. Lett. 1994;72:1451–1454.

[19] Kyprianidis IM, Stouboulos IN. Synchronization of two resistively coupled nonautonomous and hyperchaotic oscillators. Chaos Solit. Fract. 2003;17:314–325.

[20] Kyprianidis IM, Stouboulos IN. Synchronization of three coupled oscillators with ring connection. Chaos Solit. Fract. 2003;17:327–336.

http://dx.doi.org/10.5772/64811

[21] Woafo P, Enjieu Kadji HG. Synchronized states in a ring of mutually coupled self-sustained electrical oscillators. Phys. Rev. E. 2004;69:046206.

[22] Kyprianidis IM, Volos CK, Stouboulos IN, Hadjidemetriou J. Dynamics of two resistively coupled Duffing-type electrical oscillators. Int. J. Bifurcat. Chaos. 2006;16:1765–1775.

[23] Kyprianidis IM, Volos CK, Stouboulos IN. Experimental synchronization of two resistively coupled Duffing-type circuits. Nonlin. Phenom. Complex Syst. 2008;11:187–192.

[24] Dykman GI, Landa PS, Neymark YI. Synchronizing the chaotic oscillations by external force. Chaos Solit. Fract. 1991;1:339–353.

[25] Parlitz U, Junge L, Lauterborn W, Kocarev L. Experimental observation of phase synchronization. Phys. Rev. E. 1996;54:2115–2217.

[26] Rosenblum MG, Pikovsky AS, Kurths J. From phase to lag synchronization in coupled chaotic oscillators. Phys. Rev. Lett. 1997;78:4193–4196.

[27] Taherion S, Lai YC. Observability of lag synchronization of coupled chaotic oscillators. Phys. Rev. E. 1999;59:R6247–R6250.

[28] Rulkov NF, Sushchik MM, Tsimring LS, Abarbanel HDI. Generalized synchronization of chaos in directionally coupled chaotic systems. Phys. Rev. E. 1995;51:980–994.

[29] Kim CM, Rim S, Kye WH, Rye JW, Park YJ. Anti-synchronization of chaotic oscillators. Phys. Lett. A. 2003;320:39–46.

[30] Liu W, Qian X, Yang J, Xiao J. Antisynchronization in coupled chaotic oscillators. Phys. Lett. A. 2006;354:119–125.

[31] Cao LY, Lai YC. Antiphase synchronism in chaotic system. Phys. Rev. 1998;58:382–386.

[32] Astakhov V, Shabunin A, Anishchenko V. Antiphase synchronization in symmetrically coupled self-oscillators. Int. J. Bifurcat. Chaos. 2000;10:849–857.

[33] Zhong GQ, Man KF, Ko KT. Uncertainty in chaos synchronization. Int. J. Bifurcat. Chaos. 2001;11:1723–1735.

[34] Blazejczuk-Okolewska B, Brindley J, Czolczynski K, Kapitaniak T. Antiphase synchronization of chaos by noncontinuous coupling: two impacting oscillators. Chaos Solit. Fract. 2001;12:1823–1826.

[35] Kyprianidis IM, Bogiatzi AN, Papadopoulou M, Stouboulos IN, Bogiatzis GN, Bountis T. Synchronizing chaotic attractors of Chua's canonical circuit. The case of uncertainty in chaos synchronization. Int. J. Bifurcat. Chaos. 2006;16:1961–1976.

[36] Tsuji S, Ueta T, Kawakami H. Bifurcation analysis of current coupled BVP oscillators. Int. J. Bifurcat. Chaos. 2007;17:837–850.

[37] Mainieri R, Rehacek J. Projective synchronization in three-dimensional chaotic system. Phys. Rev. Lett. 1999;82:3042–3045.

[38] Voss HU. Anticipating chaotic synchronization. Phys. Rev. E. 2000;61:5115–5119.

[39] Li GH. Inverse lag synchronization in chaotic systems. Chaos Solit. Fract. 2009;40:1076–1080.

[40] Gonzalez-Miranda JM. Synchronization and control of chaos. London: Imperial College Press; 2004. 212 p.

[41] Zhan M, Hu G, Yang J. Synchronization of chaos in coupled systems. Phys. Rev. E. 2000;62:2963–2966.

[42] Wang J, Che YQ, Zhou SS, Deng B. Unidirectional synchronization of Hodgkin-Huxley neurons exposed to ELF electric field. Chaos Solit. Fract. 2009;39:1335–1345.

[43] Tass P, Rosemblum MG, Weule MG, Kurths J, Pikovsky A, Volkmann J, Schnitzler A, Freund HJ. Detection of *n*:*m* phase locking from noise data: Application to magneto-encephalography. Phys. Rev. Lett. 1998;81:3291–3294.

[44] Tognoli E, Kelso JAS. Brain coordination dynamics: True and false faces of phase synchrony and metastability. Prog. Neurobiol. 2009;87:31–40.

[45] Shaw R. Strange attractors, chaotic behavior, and information flow. Z. Nat. 1981;361:80–112.

[46] Hasselblatt B, Katok A. A first course in dynamics: with a panorama of recent developments. Cambridge: University Press; 2003. 419 p.

[47] Tacha OI, Volos ChK, Kyprianidis IM, Stouboulos IN, Vaidyanathan S, Pham V-T. Analysis, adaptive control and circuit simulation of a novel nonlinear finance system. Appl Math Comput. 2016;276:200–217.

[48] Manneville P, Pomeau Y. Intermittency and the Lorenz model. Phys Lett. 1979;75A:1–2.

[49] Grebogi C, Ott E, Yorke JA. Crises, sudden changes in chaotic attractors and chaotic transients. Phys. D. 1983;7:181–200.

[50] Rollins RW, Hunt ER. Intermittent transient chaos at interior crisis in the diode resonator. Phys Rev A. 1984;29:3327.

[51] Padmanaban E, Hens C, Dana K. Engineering synchronization of chaotic oscillator using controller based coupling design. Chaos. 2011;21:013110.

[52] Volos CK, Pham V-T, Vaidyanathan S, Kyprianidis IM, Stouboulos IN. Synchronization phenomena in coupled hyperchaotic oscillators with hidden attractors using a nonlinear open loop controller. In: Advances and applications in chaotic systems. Switzerland: Springer International Publishing; 2016. p. 1–38.

Chapter 13

Design of Dynamic Output Feedback Laws Based on Sums of Squares of Polynomials

Kenta Hoshino , Daisuke Sonoda and
Jun Yoneyama

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/64404

Abstract

We consider the stabilization of nonlinear polynomial systems and the design of dynamic output feedback laws based on the sums of squares (SOSs) decompositions. To design the dynamic output feedback laws, we show the design conditions in terms of the state-dependent linear matrix inequalities (SDLMIs). Because the feasible solutions of the SDLMIs are found by the SOS decomposition, we can obtain the dynamic output feedback laws by using numerical solvers. We show numerical examples of the design of dynamic output feedback laws.

Keywords: sums of squares polynomials, output feedback stabilization, Lyapunov methods, state-dependent LMIs

1. Introduction

In the last few decades, control design methods based on numerical methods have appeared in the control literature. Major progress in the 1980s was the emergence of numerical methods based on linear matrix inequalities (LMIs) [1]. The methods provide the numerical solutions to linear control problems in the formulation of the semidefinite programming. The LMI approach provides the design methods of feedback laws for the asymptotic stabilization, H-infinity control, and robust control. For the nonlinear control problems, the sums of squares (SOS) approach is introduced as a generalization of the LMI approach to nonlinear systems [2–6]. A feature of the sums of squares polynomials is negative semidefiniteness, and this is suitable for the stability analysis of nonlinear systems based on the Lyapunov theory. The studies [2, 3] have shown that the sums of squares decomposition can be solved in the formulation of the semidefinite programming. The result leads to the development of

numerical methods for the analysis and synthesis of nonlinear polynomial systems. Applications to control problems are feedback design [7, 8], motion planning [9], modeling, and control of fuzzy systems [10] to mention a few. Applications of the SOS approach to nonpolynomial systems are found in reference [11, 12].

The SOS approach has been the basis of numerical methods for the analysis and the synthesis of nonlinear systems. Although the Lyapunov-based approach offers the methods for the analysis and the synthesis, the construction of Lyapunov functions is often a difficult task. The SOS approach provides a technique to find Lyapunov functions by formulating the Lyapunov inequality conditions into the SOS conditions. The stability of nonlinear systems is analyzed by a direct application of SOS decompositions to the Lyapunov stability analysis. However, applications of the SOS approach to Lyapunov-based feedback design are much complicated because decision variables do not enter the Lyapunov inequalities conditions linearly. So far, two main approaches have been proposed. One is a method in [8], which formulates the design conditions into state-dependent linear matrix inequalities (SDLMIs) conditions. The SDLMIs are solved by the SOS decompositions. The other method is based on an iterative algorithm shown in reference [7], which also considers the enlargement of the regions of attraction of the closed-loop systems.

In the actual control problems, we often cannot measure all the values of the state variables of control systems. This fact leads to the necessity of the design of output feedback laws. The design of output feedback laws is more complicated task than that of state feedback laws because the stability conditions of the closed-loop systems become complex. As far as the authors know, so far, a few output feedback design methods have been proposed, for example, [[7], Section 3.5] and [13–15]. The further developments of design methods for output feedback laws have been desired.

It is well known that we often can design dynamic feedback laws even when the design of static output feedback laws is difficult. This leads to the motivation of developing a design method based on the SOS approach for the design of dynamic output feedback laws. In reference [7], an iterative method for the design of dynamic output feedback laws has been shown. However, we need to give control Lyapunov functions (CLFs) to start the iteration in the method, and this might be a difficult task especially for complex or high-dimensional systems. The state-dependent LMI approach can be an alternative approach because it does not need to give any CLF. However, a concrete method for dynamic output feedback laws has not been shown in this direction yet.

We provide the design methods of dynamic output feedback laws for the stabilization based on the SDLMI approach. This method is based on the design method of state feedback laws based on the SDLMI approach [8]. The proposed method employs a two-step algorithm. We first design a virtual state feedback law for a given system using the method of reference [8]. Then, we design a dynamic output feedback by using an SDLMI again based on the virtual state feedback law. The use of the virtual state feedback inherits the design approach of output feedback laws in reference [16], which indicates the general design approach of output feedback laws not necessarily for the SOS approach. We also show some numerical examples to demonstrate the effectiveness of the proposed method to the actual control problems.

http://dx.doi.org/10.5772/64404

Notation: We denote the set of the real numbers and integers as $\mathbb{R}$ and $\mathbb{Z}$, respectively. The notation $\mathbb{Z}_+$ is the nonnegative integers. The notation $\|x\|$ is the Euclidean norm of a vector x. For $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}_+^n$, $|\alpha|$ denotes $|\alpha| = \sum_{i=1}^{n} \alpha_i$. For a matrix $X \in \mathbb{R}^{n\times n}$, He ($X$) denoes He ($X$) = $X + X^T$.

2. Preliminary: stability of nonlinear systems

This section provides the stability theory of nonlinear systems. We present the definitions of stability, and then, we introduce the Lyapunov stability theory. The Lyapunov stability theory forms the basis for the analysis and synthesis of the stability of dynamical systems. The theory states that the existence of a kind of functions implies the stability.

This section considers the stability of an autonomous nonlinear system

$$\dot{x} = f(x), \quad x(t_0) = x_0 \tag{1}$$

where $x \in \mathbb{R}^n$ is the state, $f: \mathbb{R}^n \to \mathbb{R}^n$ is the vector fields, and $x_0 \in \mathbb{R}^n$ is the initial value of the state. In the following, we assume that the origin $x = 0$ is the equilibrium of system (1), that is, $f(0) = 0$, and we consider the stability of the origin.

To begin with, we show the definitions of the stability.

Definition 1 (stability). The equilibrium $x = 0$ is said to be Lyapunov stable if for any $\epsilon > 0$, there exists $\delta = \delta(\epsilon) > 0$, such that for any $\|x_0\| < \delta$, the solution $x(t)$ of (1) satisfies that

$$\| x(t) \| < \epsilon, \quad \forall\, t \in [t_0, \infty).$$

Definition 2 (asymptotic stability). The equilibrium $x = 0$ is said to be asymptotically stable if it is stable and there exists $\delta > 0$, such that for any $\|x_0\| < \delta$, the solution of (1) satisfies that

$$x(t) \to 0 \text{ as } t \to \infty.$$

Definition 3 (global asymptotic stability). The equilibrium $x = 0$ is said to be globally asymptotically stable if it is stable and for any $x_0 \in \mathbb{R}^n$, the solution $x(t)$ of (1) satisfies that

$$x(t) \to 0 \text{ as } t \to \infty.$$

To introduce the Lyapunov stability theory, we provide the definitions of the properties of functions.

Definition 4 (positive definiteness). A function $h: \mathbb{R}^n \to \mathbb{R}$ is said to be positive definite if $h(x) > 0$ for any $x \neq 0$ and $h(0) = 0$.

Definition 5 (positive semidefiniteness). A function $h: \mathbb{R}^n \to \mathbb{R}$ is said to be positive semidefinite if $h(x) \geq 0$ for any $x \in \mathbb{R}^n$.

We say that a function $h(x)$ is negative definite (negative semidefinite) if the function $-h(x)$ is positive definite (respectively, positive semidefinite).

Definition 6 (properness). A function $h: \mathbb{R}^n \to \mathbb{R}$ is said to be proper if for any $K \in \mathbb{R}$, the sublevel set

$$\{x \in \mathbb{R}^n \mid h(x) \leq K\}$$

is bounded.

The Lyapunov stability theory is stated as follows [17].

Theorem 1. Let U be an open subset of $\mathbb{R}^n$ which contains the origin. Suppose that a function $V: U \to \mathbb{R}$ is continuously differentiable, positive definite, and proper. The equilibrium of system (1), $x = 0$, is stable if and only if the function $V(x)$ satisfies that

$$\frac{dV}{dt}(x) = \frac{\partial V}{\partial x}(x) f(x) \leq 0, \quad \forall x \in \mathbb{R}^n.$$

Moreover, the equilibrium of system (1), $x = 0$, is asymptotically stable if and only if the function $V(x)$ satisfies that

$$\frac{dV}{dt}(x) = \frac{\partial V}{\partial x}(x) f(x) < 0, \quad \forall x \neq 0.$$

When $U = \mathbb{R}^n$, the global asymptotic stability holds.

The Lyapunov theory is used to investigate the stability of nonlinear systems. However, to investigate the stability of each system by Lyapunov theory, we need to find a Lyapunov function for it. However, to find the Lyapunov functions is often a difficult task. Further, when we try to design stabilizing feedback laws based on the Lyapunov theory, we also need to find the Lyapunov function candidates for the closed-loop systems. Therefore, we require a method to find Lyapunov functions for each nonlinear system. The SOS approach provides Lyapunov functions as solutions to the SOS conditions.

3. Sums of squares polynomials and state-dependent linear matrix inequalities

This chapter introduces some definitions and results on SOS polynomials. We also introduce that SDLMIs can be solved by the SOS decomposition.

We begin with the definitions of monomials, polynomials, and sums of squares polynomials.

http://dx.doi.org/10.5772/64404

Definition 7 (monomials). Let $z = (z_1, \ldots, z_n) \in \mathbb{R}^n$, and $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}_+^n$. A monomial of z, $m_\alpha(z)$, is a function given by

$$m_\alpha(z) = \prod_{i=1}^{n} z_i^{\alpha_i}$$

Definition 8 (polynomials). Consider monomials of z, $m_{\alpha_i}(z)$, where $\alpha_i = (\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{in}) \in \mathbb{Z}_+^n$, and $c_i \in \mathbb{R}$ for $i = 1,\ldots,m$. A polynomial of z, $f(z)$, is a function given in the form of

$$f(z) = \sum_{i=1}^{m} c_i \, m_{\alpha_i}(z).$$

The degree of polynomial $f(z)$, d, is given by

$$d = \max_i |\alpha_i|.$$

Let $\mathcal{R}_n$ denote the set of polynomials of n variables. Then, we show the definition of the sums of squares polynomials.

Definition 9 (sums of squares polynomials, SOSs). Let $z = (z_1,\ldots,z_n)$. A sum of squares polynomial $\sigma_n(z)$ is a function given in the form of

$$\sigma_n(z) = \sum_{i=1}^{m} f_i(z)^2, \quad f_i(z) \in \mathcal{R}_n, \qquad i = 1, \ldots, m.$$

The decomposition of given polynomials into SOSs is called as the SOS decomposition. Regarding the SOS decomposition, the following result is shown.

Theorem ([2, 3]). Consider the polynomial of z of degree $2d$, $f(z)$. The polynomial $f(z)$ is an SOS polynomial if and only if there exist a column vector $X(z)$ whose elements are monomials of z of degree no greater than d and a positive semidefinite matrix Q such that

$$f(z) = X(z)^T Q X(z)$$

holds.

We show a simple example of SOSs.

Example 1. Consider a polynomial $f(z)$ given by

$$f(z) = z^2 + 2z + 2,$$

where $z \in \mathbb{R}$. Apparently, this polynomial is expressed as the sum of squares polynomial

$$f(z) = z^2 + 2z + 2 = (z+1)^2 + 1.$$

Regarding Theorem 2, the polynomial is also expressed as

$$f(z) = \begin{bmatrix} z & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} z \\ 1 \end{bmatrix}, \tag{2}$$

and the matrix in the right-hand side of (2) is positive definite.

The SOS decomposition can be solved by some numerical solvers, such as YALMIP [18] and SOSTOOLS [19]. When some coefficients of polynomials are decision variables in an SOS decomposition, by using the numerical solvers, we can find the feasible solutions such that the SOS decomposition holds. Therefore, we can adapt the SOS decomposition to the design of feedback laws in control problems.

With the relation to the stability theory presented in Section 2, the sufficient condition of the stability is given as the SOS conditions.

Theorem 3. [2] Consider system (1). If there exist a positive definite function $\phi: \mathbb{R}^n \to \mathbb{R}$ and an SOS polynomial $\epsilon: \mathbb{R}^n \to \mathbb{R}^n$, such that

$$\begin{aligned} \phi(x) - \epsilon(x) &> 0, \\ \frac{\partial \phi}{\partial x}(x) f(x) + \epsilon(x) &< 0, \qquad \forall\, x \neq 0 \end{aligned}$$

then the equilibrium $x = 0$ is asymptotically stable.

Theorem 3 shows a direct application of the SOSs to the analysis of the stability. This implies that the SOS decomposition can be applied to the synthesis of the stabilizing feedback laws. This chapter develops a method to design dynamic output feedback laws based on the SDLMI approach [8]. The SDLMI is defined as the optimization problem:

$$\begin{aligned} &\text{minimize} \quad \sum_{i=1}^{m} a_i c_i \\ &\text{subject to} \quad F_0(z) + \sum_{i=1}^{m} c_i F_i(z) \geq 0, \quad \text{for } \forall\, z \in \mathbb{R}^n, \end{aligned}$$

where $a_i \in \mathbb{R}$ are the fixed coefficients, c_i are the decision variables, the matrix functions $F_i: \mathbb{R}^n \to \mathbb{R}^{q \times q}$ are state-dependent symmetric matrices. The constraint should be satisfied for any $z \in \mathbb{R}^n$. This differs from standard LMIs and is the derivation of the word, state-dependent.

A relation of the SDLMIs and the SOS decompositions is shown as follows.

http://dx.doi.org/10.5772/64404

Theorem 4. ([8]) Let $d > 0$ and $F:\mathbb{R}^n \to \mathbb{R}^{q\times q}$ a symmetric polynomial matrix the elements of which are polynomials of z with degree $2d$. Moreover, consider a vector $v \in \mathbb{R}^q$. If $v^T F(z)v$ is a sum of squares polynomial, then $F(z) \geq 0$ holds for any $z \in \mathbb{R}^q$.

Theorem 4 states that if we find that the polynomial $v^T F(z)v$ is decomposed into an SOS with respect to (z,v), it implies the positive definiteness of $F(z)$ for any $z \in \mathbb{R}^n$. We can derive stability conditions in terms of SDLMIs. This leads to the design of feedback laws for the stabilization based on the combination of the SDLMIs and the SOS decomposition. We develop the synthesis of dynamic output feedback laws based on Theorem 4 in the following sections.

4. Problem setting: stabilization through dynamic output feedback

This chapter considers the stabilization problem via dynamic output feedback laws and the synthesis of the stabilizing feedback laws. This section states the problem setting.

The approach presented here is based on the SDLMI approach, which derives the sufficient conditions of the existence of stabilizing feedback laws as the SDLMI conditions. We can obtain stabilizing feedback control laws and Lyapunov functions by solving the SDLMI conditions using numerical solvers.

Consider a nonlinear system given as

$$\begin{aligned} \dot{x} &= f(x,u), \qquad x(t_0) = x_0 \\ y &= h(x), \end{aligned} \tag{3}$$

where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^{n_u}$ is the input, $y \in \mathbb{R}^{n_y}$ is the output, f: $\mathbb{R}^n \times \mathbb{R}^{n_u} \to \mathbb{R}^n$, h: $\mathbb{R}^n \to \mathbb{R}^{n_y}$, and x_0 is the initial state. For the nonlinear systems given by (3), we assume that system (3) is expressed as

$$\begin{aligned} \dot{x} &= A(x)Z(x) + B(x)u, \quad x(t_0) = x_0 \\ y &= C(x)Z(x), \end{aligned} \tag{4}$$

where Z: $\mathbb{R}^n \to \mathbb{R}^N$, A: $\mathbb{R}^n \to \mathbb{R}^{n\times N}$, B: $\mathbb{R}^n \to \mathbb{R}^{n\times n_u}$, C: $\mathbb{R}^n \to \mathbb{R}^{n_y\times N}$. Further, we assume that $Z(x) = 0$, if and only if $x = 0$. We consider the output stabilization of system (4) using a dynamic feedback law in the form of

$$\begin{aligned} \dot{\hat{x}} &= A_c\left(\hat{x},y\right)\hat{x} + B_c\left(\hat{x},y\right)y, \qquad \hat{x}\left(t_0\right) = \hat{x}_0 \\ u &= C_c\left(\hat{x},y\right)\hat{x} + D_c\left(\hat{x},y\right)y, \end{aligned} \tag{5}$$

where $\hat{x} \in \mathbb{R}^{n_{\hat{x}}}$ is the state of the dynamic feedback law,

$A_c: \mathbb{R}^{n_{\hat{x}}} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_{\hat{x}} \times n_{\hat{x}}}, B_c: \mathbb{R}^{n_{\hat{x}}} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_{\hat{x}} \times n_y}, C_c: \mathbb{R}^{n_{\hat{x}}} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_u \times n_{\hat{x}}}$, and $D_c: \mathbb{R}^{n_{\hat{x}}} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_u \times n_y}$, and $\hat{x}_0$ is the initial state.

We have the closed-loop system of (4) with the dynamic output feedback law (5), given by

$$\begin{aligned} \dot{x} &= \left\{A(x) + B(x)D_c(\hat{x}, y)C(x)\right\}Z(x) + B(x)C_c(\hat{x}, y)\hat{x}, \\ \dot{\hat{x}} &= A_c(\hat{x}, y)\hat{x} + B_c(\hat{x}, y)C(x)Z(x). \end{aligned} \tag{6}$$

We consider the stabilization of the closed-loop system (6). To this end, we give a method to design the matrix functions $A_c(\hat{x}, y), B_c(\hat{x}, y), C_c(\hat{x}, y), D_c(\hat{x}, y)$ in the next section.

Remark 1. We obtain a system in the form of (4) as an expression of a nonlinear affine system

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

by choosing *Z(x)* properly. Note that the choice of *Z(x)* is not unique in general. The systems in the form of (4) can be seen as a generalization of linear systems, given as

$$\begin{aligned} \dot{x} &= Ax + Bu, \\ y &= Cx, \end{aligned}$$

where the matrices *A*, *B*, and *C* are with the appropriate dimensions.

5. Design of dynamic output feedback laws through SOSs

This section provides a design method of dynamic feedback laws (5) for the output stabilization of system (4). We show stability conditions of the closed-loop system of (6) as SDLMI conditions. We can obtain the stabilizing laws by solving the SDLMI conditions via SOS decomposition using numerical solvers.

The main idea of the proposed method is as follows. Instead of the dynamic feedback law (5), assume that there exists a static state feedback law

$$u = k(x), \tag{7}$$

where $k: \mathbb{R}^n \to \mathbb{R}^{n_u}$, such that the feedback law asymptotically stabilizes the origin of system (4). Then, according to the converse Lyapunov theorem, we have a Lyapunov function $U_1(x)$. Then, we consider the design of the dynamic output feedback law (5) so that a function $U(x, \hat{x})$ given by

http://dx.doi.org/10.5772/64404

$$U(x,\hat{x}) = U_1(x) + (k(x)-\hat{x})^T\Sigma(k(x)-\hat{x}) \tag{8}$$

becomes the Lyapunov function of the closed-loop system (6) with some positive definite matrix Σ. When we design the output feedback laws, so that the function $U(x,\hat{x})$ of (8) is a Lyapunov function of the closed-loop system, the value of $\hat{x}$ of the designed output feedback laws in (8) will estimate the value of $k(x)$. A design procedure discussed here can be seen in reference [16], and is called as the direct design. As shown in the following, when we obtain the static feedback law (7) in polynomial forms, we can obtain the SDLMI conditions where the stability of the closed-loop system (6) is guaranteed by function (8).

In the following, if the matrix $B(x)$ of (4) has rows all the elements of which are zero, we denote the corresponding row indices as $J = \{j_1, \cdots, j_p\}$. We also employ the notation $\tilde{x} = (x_{j_1}, x_{j_2}, \cdots, x_{j_p})$.

As discussed above, we design a stabilizing state feedback law as the first step. The state feedback law also can be designed by using SDLMIs. We introduce the following result shown in reference [8].

Theorem 5. ([8]) Suppose that there exist a symmetric polynomial matrix P: $\mathbb{R}^n \to \mathbb{R}^{N\times N}$, a polynomial matrix K: $\mathbb{R}^n \to \mathbb{R}^{n_u\times N}$, a parameter $\epsilon_1 > 0$, and an SOS polynomial ϵ_2: $\mathbb{R}^n \to \mathbb{R}$, such that

$$\begin{aligned} &v^T(P(\tilde{x})-\epsilon_1 I)v, \\ &-v\Big\{P(\tilde{x})A(x)^TM(x)^T + M(x)A(x)P(\tilde{x}) + K(x)^TB(x)^TM(x)^T \\ &\qquad + M(x)B(x)K(x) - \sum_{j\in J}\frac{\partial P}{\partial x_j}(\tilde{x})\big(A_j(x)Z(x)\big) + \epsilon_2(x)I\Big\}v \end{aligned} \tag{9}$$

are SOS polynomials, where $v \in \mathbb{R}^N$, $A_j(x)$ is the j-th row of A(x), and

$$M(x) = \frac{\partial Z}{\partial x}(x). \tag{10}$$

Then, the origin of (4) is asymptotically stabilized by a state feedback given by

$$u = k(x) = K(x)P(\tilde{x})^{-1}Z(x). \tag{11}$$

For the design of the output feedback laws, we show the following theorem as the main result, which gives a design condition of the feedback law (5) in terms of state-dependent matrix inequalities.

Theorem 6. Suppose that there exist a symmetric matrix $P_1 \in \mathbb{R}^{N\times N}$, a polynomial matrix K: $\mathbb{R}^n \to \mathbb{R}^{n_u\times N}$, a parameter $\epsilon_1 > 0$, and an SOS polynomial ϵ_2: $\mathbb{R}^n \to \mathbb{R}$, such that

$$v^T(P_1 - \epsilon_1 I)v,$$
$$-v^T\{P_1A(x)^TM(x)^T + M(x)A(x)P_1 + K(x)^TB(x)^TM(x)^T + M(x)B(x)K(x) + \epsilon_2(x)I\}v \tag{12}$$

are SOS polynomials, where $v \in \mathbb{R}^N$ and $M(x)$ is given as 10. Further, suppose that there exist a symmetric matrix $P_2 : \mathbb{R}^{(N+n_{\hat{x}}) \times (N+n_{\hat{x}})}$, and an SOS polynomial $\epsilon_3 : \mathbb{R}^n \times \mathbb{R}^{n_{\hat{x}}} \to \mathbb{R}$, such that

$$-w^T\left(\begin{pmatrix} \Lambda_{11}(x,\hat{x}) & \Lambda_{12}(x,\hat{x}) \\ \Lambda_{12}^T(x,\hat{x}) & \Lambda_{22}(x,\hat{x}) \end{pmatrix} + \epsilon_3(x,\hat{x})I\right)w \tag{13}$$

is an SOS polynomial where $w \in \mathbb{R}^{N+n_{\hat{x}}}$, and

$$\begin{aligned}
\Lambda_{11}(x,\hat{x}) &= \mathrm{He}(P_1^{-1}M(x)(A(x) + B(x)D_c(\hat{x},y)C(x)) \\
&\quad + P_1^{-1}K(x)^T P_2\left(\left\{\frac{\partial k}{\partial x}(x)(A(x) + B(x)D_c(\hat{x},y)C(x)) - B_c(\hat{x},y)C(x)\right\}\right), \\
\Lambda_{12}(x,\hat{x}) &= P_1^{-1}M(x)B(x)C_c(\hat{x},y) + P_1^{-1}K(x)^T P_2\left(\frac{\partial k}{\partial x}(x)B(x)C_c(\hat{x},y) - A_c(x)\right) \\
&\quad + \left\{B_c(\hat{x},y)C(x) - \frac{\partial k}{\partial x}(x)(A(x) + B(x)D_c(\hat{x},y)C(x))\right\}^T P_2, \\
\Lambda_{22}(x,\hat{x}) &= \mathrm{He}\left(P_2\left(A_c(\hat{x},y) - \frac{\partial k}{\partial x}(x)B(x)C_c(\hat{x},y)\right)\right)
\end{aligned}$$

where the matrices $A_c(\hat{x},y)$, $B_c(\hat{x},y)$, $C_c(\hat{x},y)$, *and* $D_c(\hat{x},y)$, are given in (5). Then, the dynamic output feedback law (5) globally asymptotically stabilizes the origin of the system (4).

Proof. According to Theorem 6, the function

$$U_1(x) = Z(x)^T P_1^{-1} Z(x)$$

is the Lyapunov function of the closed-loop system of (4) with the state feedback law

$$u = k(x) = K(x)P_1^{-1}Z(x).$$

Then, to consider a dynamic output feedback law in the form of (5), we consider a function given by

$$V(x,\hat{x}) = U_1(x) + U_2(x,\hat{x}), \tag{14}$$

where the function $U_2(x,\hat{x})$ is given by

http://dx.doi.org/10.5772/64404

$$U_2(x,\hat{x}) = (k(x)-\hat{x})^T P_2 (k(x)-\hat{x}).$$

Then, the time derivative of function (14) along the trajectory of the closed-loop system (6) is given as

$$V(x,\hat{x}) = U_1(x) + U_2(x,\hat{x}),$$

where

$$\begin{aligned} U_1(x) &= Z(x)^T P_1^{-1} Z(x) + Z(x)^T P_1^{-1} Z(x) \\ &= Z(x)^T \mathrm{He}\left(P_1^{-1} M(x)\left\{A(x) + B(x) D_c(\hat{x},y) C(x)\right\}\right) Z(x) \\ &+ \hat{x}^T C_c(\hat{x},y)^T B(x)^T M(x)^T P_1^{-1} Z(x) \\ &+ Z(x)^T P_1^{-1} M(x) B(x) C_c(\hat{x},y)\hat{x}, \end{aligned} \tag{15}$$

and

$$\begin{aligned} \dot{U}_2(x,\hat{x}) &= \left(\dot{k}(x)-\dot{\hat{x}}\right)^T P_2 (k(x)-\hat{x}) + (k(x)-\hat{x})^T P_2 \left(\dot{k}(x)-\dot{\hat{x}}\right) \\ &= Z(x)^T He\left(P_1^{-1} K(x)^T P_2 \left\{\frac{\partial k}{\partial x}(x)\left(A(x)+B(x)D_c(\hat{x},y)C(x)\right) - B_c(\hat{x},y)C(x)\right\}\right) Z(x) \\ &+ Z(x)^T \left[\left\{-\frac{\partial k}{\partial x}(x)\left(A(x)+B(x)D_c(\hat{x},y)C(x)\right) + B_c(\hat{x},y)C(x)\right\}^T P_2\right. \\ &\left. + P_1^{-1} K(x)^T P_2 \left(\frac{\partial k}{\partial x}(x) B(x) C_c(\hat{x},y) - A_c(\hat{x},y)\right)\right] \hat{x}^T \\ &+ \hat{x}^T \left[\left\{\frac{\partial k}{\partial x}(x) B(x) C_c(\hat{x},y) - A_c(\hat{x},y)\right\}^T P_2 K(x) P_1^{-1}\right. \\ &\left. + P_2\left\{-\frac{\partial k}{\partial x}(x)\left(A(x)+B(x)D_c(\hat{x},y)C(x)\right) + B_c(\hat{x},y)C(x)\right\}\right] Z(x) \\ &+ \hat{x}^T \mathrm{He}\left(P_2\left(A_c(\hat{x},y) - \frac{\partial k}{\partial x}(x) B(x) C_c(\hat{x},y)\right)\right)\hat{x}. \end{aligned} \tag{16}$$

Therefore, the time derivative of the function $V(x,\hat{x})$ along the solution of system (6) is given as

$$\begin{aligned} \dot{V}(x,\hat{x}) &= \dot{U}_1(x) + \dot{U}_2(x,\hat{x}) \\ &= Z(x)^T \Lambda_{11}(x,\hat{x}) Z(x) + Z(x)^T \Lambda_{12}(x,\hat{x})\hat{x} + \hat{x}^T \Lambda_{12}(x,\hat{x})^T Z(x) \\ &+ \hat{x}^T \Lambda_{22}(x,\hat{x})\hat{x} \\ &= \begin{bmatrix} Z(x)^T & \hat{x}^T \end{bmatrix} \begin{bmatrix} \Lambda_{11}(x,\hat{x}) & \Lambda_{12}(x,\hat{x}) \\ \Lambda_{12}(x,\hat{x})^T & \Lambda_{22}(x,\hat{x}) \end{bmatrix} \begin{bmatrix} Z(x) \\ \hat{x} \end{bmatrix}. \end{aligned} \tag{17}$$

Then, condition (13) of the theorem and Theorem 4 imply that

$$\begin{bmatrix} \Lambda_{11}(x,\hat{x}) & \Lambda_{12}(x,\hat{x}) \\ \Lambda_{12}(x,\hat{x})^T & \Lambda_{22}(x,\hat{x}) \end{bmatrix} < 0, \quad \forall (x,\hat{x}) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_x}. \tag{18}$$

From (17) and (18), we can conclude that $\dot{V}(x,\hat{x})$ is negative definite. Therefore, according to Theorem 1, we can conclude that the origin of the closed-loop system is globally asymptotically stable. This completes the proof.

When we design the dynamic output feedback law (5) according to Theorem 6, we first solve the SOS decomposition of condition (12) to find the matrix P_1. Then, if we can obtain the feasible solutions of the matrix P_1 and the function $K(x)$ satisfying condition (12), we try to find the matrix functions $A_c(\hat{x},y)$, $B_c(\hat{x},y)$, $C_c(\hat{x},y)$, $D_c(\hat{x},y)$, the matrix P_2, and the SOS polynomial $\epsilon_3 > 0$ satisfying condition (13). At this time, because the decision variables do not enter in (13) linearly, we set $P_2 = I$ in general. Then, we can consider the SOS decomposition for (13). If we can find the feasible solution of condition (13), we will obtain the stabilizing feedback laws in the form of (5).

Remark 2. The condition of (12) in Theorem 6 corresponds to the condition of (9) in Theorem 5. Note that the matrix P_1 in Theorem 6 is a constant matrix, although the matrix $P(x)$ in Theorem 5 is the function of x. This is due to the fact that the inverse of the matrix P_1 appears in (16). If the matrix P_1 is the polynomial matrix in Theorem 6, we cannot employ the SOS decomposition. Therefore, we limit ourselves to the case of the constant matrices in Theorem 6.

6. Numerical examples of dynamic output feedback stabilization

6.1. Numerical example 1

This section shows some numerical examples of the dynamic output feedback stabilization by the proposed method shown in Section 5.

We show the first example of the stabilization. Consider a system given by

$$\begin{aligned} \dot{x}_1 &= 0.5\,x_1 - 0.1x_1^3 + u, \\ \dot{x}_2 &= x_1^2 - x_2, \\ y &= x_1, \end{aligned} \tag{19}$$

where $x = (x_1, x_2)^T$ is the state, $y \in \mathbb{R}$ is the output, and $u \in \mathbb{R}$ is the input. In order to design a dynamic output feedback law for the stabilization of system (19) based on the result presented in the previous section, we choose $Z(x)=(x_1, x_2)^T$. Then, we have the expression of system (19) in the form of (4), where

$$A(x) = \begin{pmatrix} 0.5 - 0.1x_1^2 & 0 \\ x_1 & -1 \end{pmatrix}, \quad B(x) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad C(x) = (1 \quad 0).$$

http://dx.doi.org/10.5772/64404

We consider the output feedback stabilization of system (19) using the dynamic feedback law (5). We consider a low-dimensional dynamic feedback, and we assume that $n_{\hat{x}} = 1$. According to Theorem 6, by choosing $P_2 = I$, we obtained the matrix P_1 and the function $K(x)$ by solving the SOS decomposition of (12) using YALMIP. We consider the function $K(x)$ with zero degree. The obtained matrix P_1 and the function $K(x)$ are given as

$$P_1 = \begin{bmatrix} 1.2306\times10^{-2} & -9.8824\times10^{-11} \\ -9.8824\times10^{-11} & 5.2061\times10^{-2} \end{bmatrix},$$
$$K(x) = \begin{bmatrix} -6.9660\times10^{-3} & -4.9775\times10^{-6} \end{bmatrix}.$$

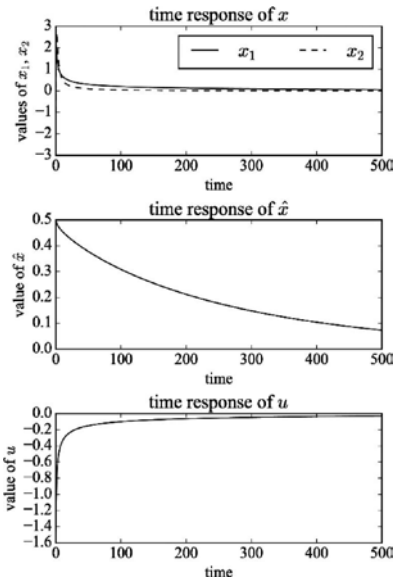

Figure 1. Time responses of x, $\hat{x}$, and u of (19) with dynamic output feedback law (5) with degree zero one.

Then, by using P_1 and $K(x)$, we found the feasible solution $A_c(\hat{x},y), B_c(\hat{x},y), C_c(\hat{x},y), D_c(\hat{x},y)$, which are two degree, to the SOS decomposition of condition (13). Therefore, we obtain the dynamic output feedback laws that stabilizes system (19), given by

$$\begin{aligned}
A_c(\hat{x},y) &= -0.003412109272 + 1.3668\times10^{-5}\,y - 0.0034187\,y^2 + 1.4670\times10^{-5}\,\hat{x} \\
&\quad - 2.3942\times10^{-5}\,y\,\hat{x} - 0.0034045\,\hat{x}^2, \\
B_c(\hat{x},y) &= -6.91051768\times10^{-5} + 1.4074\times10^{-6}\,y + 1.6941\times10^{-5}\,y^2 - 5.1525\times10^{-6}\,\hat{x} \\
&\quad + 7.2076\times10^{-6}\,y\,\hat{x} - 1.1067\times10^{-5}\,\hat{x}^2, \\
C_c(\hat{x},y) &= 9.38813958\times10^{-6} + 3.7173\times10^{-5}\,y + 1.9926\times10^{-4}\,y^2 - 5.5249\times10^{-6}\hat{x} \\
&\quad + 1.3925\times10^{-4}\,y\,\hat{x} + 3.5555\times10^{-6}\,\hat{x}^2, \\
D_c(\hat{x},y) &= -0.5020898379 - 3.4848\times10^{-4}\,y + 0.0023489\,y^2 - 2.2606\times10^{-5}\,\hat{x} \\
&\quad + 2.1822\times10^{-4}\,y\,\hat{x} - 0.0068477\,\hat{x}^2.
\end{aligned} \tag{20}$$

Figure 1 shows the time responses of the state variables $x(t)$, $\hat{x}(t)$ and $u(t)$ of the closed-loop system (19) with the designed dynamic output feedback (20). The initial values are chosen as $x(0) = (3,-1)$,and $\hat{x}(0) = 0.5$. In **Figure 1**, the states $x(t)$ and $\hat{x}(t)$ converge to the origin.

Then, we also obtain a dynamic output feedback control law in the case where the elements of $K(x)$ are degree zero, and the elements of $A_c(\hat{x},y), B_c(\hat{x},y), C_c(\hat{x},y)$, and $D_c(\hat{x},y)$ are degree three with respect to $\hat{x}$ and y. Again, by solving the SOS decomposition following Theorem 6, we obtain the value of the matrix P_1 and the function $K(x)$ as same as above.

We also obtain the values of $A_c(\hat{x},y), B_c(\hat{x},y), C_c(\hat{x},y)$, and $D_c(\hat{x},y)$ as

$$
\begin{aligned}
A_c(\hat{x},y) = {} & -0.04345107574 - 1.9588 \times 10^{-7}\, y - 0.044197\, y^2 - 8.8208 \times 10^{-8}\, \hat{x} \\
& -4.5391 \times 10^{-7}\, y\, \hat{x} - 0.043558\, \hat{x}^2 - 2.1533 \times 10^{-5}\, y^3 + 3.9213 \times 10^{-9}\, y^2 \hat{x} \\
& -1.5638 \times 10^{-5}\, y\, \hat{x}^2 - 2.1580 \times 10^{-9}\, \hat{x}^3, \\
B_c(\hat{x},y) = {} & -9.486666512 \times 10^{-5} + 2.8266 \times 10^{-7}\, y + 1.2691 \times 10^{-4}\, y^2 - 4.6792 \times 10^{-7}\, \hat{x} \\
& -8.6168 \times 10^{-7}\, y\, \hat{x} + 2.8204 \times 10^{-5}\, \hat{x}^2 - 0.0031033\, y^3 + 5.6498 \times 10^{-7}\, y^2\, \hat{x} \\
& -0.0022537\, y\, \hat{x}^2 - 3.1099 \times 10^{-7} \hat{x}^3, \\
C_c(\hat{x},y) = {} & 0.0001750991329 - 3.9137 \times 10^{-7}\, y + 9.3325 \times 10^{-4} y^2 \\
& +1.5899 \times 10^{-8}\, \hat{x} + 4.6876 \times 10^{-8}\, y\, \hat{x} + 3.3911 \times 10^{-4}\, \hat{x}^2 + 3.8040 \times 10^{-5}\, y^3 \\
& -6.9264 \times 10^{-9}\, y^2\, \hat{x} + 2.7626 \times 10^{-5}\, y\, \hat{x}^2 + 3.8121 \times 10^{-9}\, \hat{x}^3, \\
D_c(\hat{x},y) = {} & -0.5184067865 - 2.2545 \times 10^{-7}\, y - 0.023064\, y^2 - 1.8349 \times 10^{-8}\, \hat{x} \\
& +3.8378 \times 10^{-9}\, y\, \hat{x} - 0.034449\, \hat{x}^2 + 2.1533 \times 10^{-5}\, y^3 - 3.9199 \times 10^{-9}\, y^2\, \hat{x} \\
& +1.5638 \times 10^{-5}\, y\, \hat{x}^2 + 2.1580 \times 10^{-9}\, \hat{x}^3.
\end{aligned}
$$

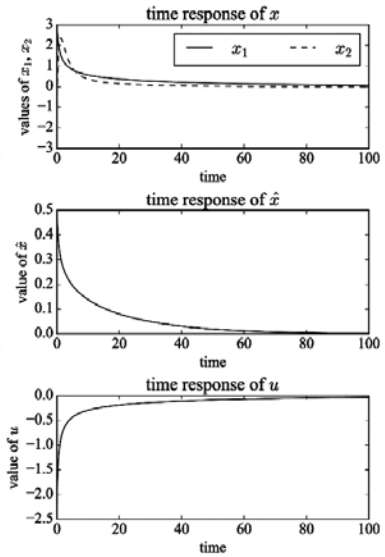

Figure 2. Time responses of x, $\hat{x}$, and u of (19) with dynamic output feedback law (5) with degree zero one.

http://dx.doi.org/10.5772/64404

The obtained feedback control law also stabilizes system (19). **Figure 2** shows the time responses of the state $x(t)$, $\hat{x}(t)$ and the input $u(t)$ of the closed-loop systems with the initial values $x(0) = (3,-1)$, and $\hat{x}(0) = 0.5$. The state converges to the origin, and the value of $u(t)$ also converges to zero.

6.2. Numerical example 2

We consider the following example, which models a circuit with negative-resistance oscillator, taken from reference [17] and modified. Consider a system given by

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -x_1 + x_2 - \tfrac{1}{3}x_2^3 + u, \\ y &= x_2, \end{aligned} \tag{21}$$

where $x = (x_1, x_2)^T$ is the state, $u \in \mathbb{R}$ is the input, and y is the output. To design the dynamic output feedback laws, we express system (21) of form (4) as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} -x_1 - \dfrac{1}{3}x_2^3 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u,$$

$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} -x_1 - \dfrac{1}{3}x_2^3 \\ x_2 \end{bmatrix}.$$

Following the design procedure in the previous section, we design the dynamic feedback control law with $n_{\hat{x}} = 1$.First, we obtain the constant matrix P_1 and the polynomial matrix $K(x)$ with degree zero. The matrix P_1 and $K(x)$ with zero degree are obtained as

$$P_1 = \begin{bmatrix} 0.0155674 & 0.0012554 \\ 0.0012554 & 0.0155125 \end{bmatrix}$$
$$K(x) = \begin{bmatrix} -0.0013102 & -0.0167686 \end{bmatrix}$$

Then, we solve the SOS decomposition (13) to find the matrices $A_c(\hat{x}, y), B_c(\hat{x}, y), C_c(\hat{x}, y)$, and $D_c(\hat{x}, y)$ with degree one. By choosing $P_2 = I$, the feasible solutions are obtained as

$$\begin{aligned} A_c(\hat{x}, y) &= -0.1675653629 - 9.5745 \times 10^{-9} y + 8.1304 \times 10^{-11} \hat{x}, \\ B_c(\hat{x}, y) &= -0.1582768369 + 5.2303 \times 10^{-8} y - 3.9643 \times 10^{-10} \hat{x}, \\ C_c(\hat{x}, y) &= 6.655333476 \times 10^{-5} + 3.0370 \times 10^{-11} y + 3.9639 \times 10^{-12} \hat{x}, \\ D_c(\hat{x}, y) &= -1.081399695 - 4.5067 \times 10^{-11} y + 4.1856 \times 10^{-12} \hat{x}. \end{aligned}$$

Figure 3 shows the time responses of the states x, $\hat{x}$ and the input u of the closed-loop systems. The figure shows that the states x and $\hat{x}$converge to the origin. Also, the figure shows that the input values converge to zero as the states converge to the origin.

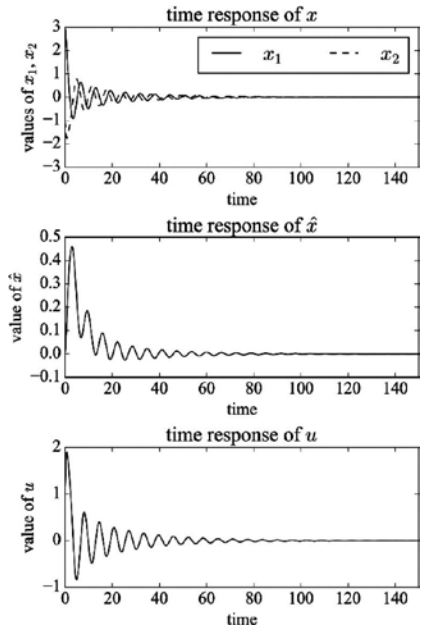

Figure 3. Time responses of x, $\hat{x}$, and u of (21) with dynamic output feedback law (5) with degree zero one.

7. Conclusion

We considered the design of dynamic output feedback laws via the SOS decomposition. For the design of the feedback laws, we derived the design conditions as the state-dependent matrix inequalities. According to the derived conditions, we can design the stabilizing feedback laws as the feasible solutions to the SDLMIs by using the numerical solvers. Future works include to derive less conservative conditions and to develop design methods of dynamic output feedback laws for advanced control, such as H-infinity control.

Author details

Kenta Hoshino*, Daisuke Sonoda and Jun Yoneyama

*Address all correspondence to: hoshino@ee.aoyama.ac.jp

Aoyama Gakuin University, Fuchinobe, Sagamihara city, Kanagawa, Japan

http://dx.doi.org/10.5772/64404

References

[1] Boyd SP, Ghaoui LE, Feron E, and Balakrishnan V. Linear matrix inequalities in system and control theory. Philadelphia, Pennsylvania: SIAM; 1994.

[2] Parrilo PA. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization [dissertation]. California Institute of Technology; 2000.

[3] Parrilo PA. Semidefinite programming relaxations for semialgebraic problems. Mathematical Programming. 2003; 96(2):294–320.

[4] Papachristodoulou A, Prajna S. A tutorial on sum of squares techniques for systems analysis. In: Proceedings of the 2005 American Control Conference; IEEE; 2005. pp. 2686–2700.

[5] Henrion D, Garulli{Editors PT} A, editors. Positive polynomials in control. Berlin Heidelberg: Springer-Verlag; 2005.

[6] Chesi G. LMI techniques for optimization over polynomials in control: a survey. IEEE Transactions on Automatic Control. 2010;55(11):2500–2510.

[7] Jarvis-Wloszek ZW. Lyapunov based analysis and controller synthesis for polynomial systems using sum-of-squares optimization [dissertation]. University of California, Berkeley: 2003.

[8] Prajna S, Papachristodoulou A, and Wu F. Nonlinear control synthesis by sum of squares optimization: a Lyapunov-based approach. In: Proceedings of the 5th Asian Control Conference; July 2004; Melbourne, Australia; 2004. pp. 157–165.

[9] Tedrake R, Manchester IR, Tobenkin M, and Roberts JW. LQRtrees: feedback motion planning via sums-of-squares verification. The International Journal of Robotics Research. 2010; 29(8): 1038–1052.

[10] Tanaka K, Yoshida H, Ohtake H, and Wang HO. A sum-of-squares approach to modeling and control of nonlinear dynamical systems with polynomial fuzzy systems. IEEE Transactions on Fuzzy Systems. 2009; 17(4): 911–922.

[11] Papachristodoulou A, and Prajna S. Analysis of non-polynomial systems using the sum of squares decomposition. In: Henrion D, and Garulli A, editors. Positive polynomials in control. Berlin, Heidelberg: Springer; 2005. pp. 23–43.

[12] Jarvis-Wloszek Z, Feeley R, Tan W, Sun K, and Packard A. Control applications of sum of squares programming. In: Henrion D, Garulli A, editors. Positive polynomials in control. Berlin, Heidelberg: Springer-Verlag; 2005. pp. 3–22.

[13] Henrion D, and Lasserre JB. Convergent relaxations of polynomial matrix inequalities and static output feedback. IEEE Transactions on Automatic Control. 2006; 51(2):192–202.

[14] Zhao D. and Wang J-L. Robust static output feedback design for polynomial nonlinear systems. International Journal of Robust and Nonlinear Control. 2010;20(14):1637–1654.

[15] Chesi G. Computing output feedback controllers to enlarge the domain of attraction in polynomial systems. IEEE Transactions on Automatic Control. 2004;49(10):1846–1853.

[16] Andrieu V, Praly L. A unifying point of view on output feedback designs for global asymptotic stabilization. Automatica. 2009; 45(8):1789–1798.

[17] Khalil HK. Nonlinear systems. 3rd ed. Upper Saddle River: Prentice-Hall; 2002.

[18] Löfberg J. YALMIP: A toolbox for modeling and optimization in. In: CACSD Conference; Taipei, Taiwan. 2004.

[19] Prajna S, Papachristodoulou A, Parrilo P. Introducing SOSTOOLS: A general purpose sum of squares programming solver. In: the 41st IEEE Conference on Decision and Control; 2002; Las Vegas, USA; 2002. pp. 741–746.

Chapter 14

Could the Stock Return be a Leading Indicator of the Economic Growth in the Depression? Analysis Based on Nonlinear Dynamic Panel Model

Lee Yuan-Ming and Wang Kuan-Min

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/63629

Abstract

This chapter examines whether the stock return can be a leading indicator of economic growth in the depression. A nonlinear dynamic panel data model is constructed with the use of the new current depth of recession (NCDR) indicator as the regime switched factor. Our findings show that in the recession period, the stock return can significantly explain the economic growth. As to the impact of a country's development level and business cycle stages, the stock return can serve as a leading indicator of the economic growth in the Asian emerging markets in the recovery subperiod.

Keywords: stock returns, economic growth, dynamic panel data model, recession period, business cycle

1. Introduction

Fluctuations of stock returns are highly correlated with economic activities. Generally speaking, the reason why the stock market is called the economy showcase is that the current-period stock return could be viewed as a leading indicator of the economic growth in the future. To some extent, most stock markets are efficient. In an efficient stock market, the current stock price reveals the discounted value of future dividends and capital returns. The current stock price also reflects the investors' expectations of the future of the economy. Therefore, from the macroeconomics view point, the stock market reveals the economic trend of the country. On the field studying the relationship between the stock return and economic growth, in the beginning, most researches use countries such as United States [1, 2], Canada [3, 4], or G7 countries [5, 6] as samples. Later on, members of European Union, members of the Organization

for Economic Cooperation and Development (OECD), or the Asian emerging markets (such as Singapore, Korea, the Philippines, Malaysia, Indonesia, and Taiwan) are included as sample countries as well. Research examples include in [7–11]. Most of these studies find that the relationship between the stock return and economic growth is statistically significant.

The main focus of this study is to investigate whether the relationship between the stock return and economic growth would be the same for countries with different development levels in the depression and recovery subperiods.[1] This study employs a 26-country sample; the sample period is from the first quarter of 1982 to the fourth quarter of 2009, longer than that of [9].[2] The new current depth of recession (NCDR) indicator proposed by Bradley and Jansen [12] is used here as the switched factors to capture the subperiods in the recession period.

In researches using the nonlinear model to study the relationship between the stock return and economic growth, authors often construct two regimes to divide the business cycle into two periods, the expansionary and recession periods. For example, Henry et al. [9] argue that output is characterized by asymmetry—the so-called bounce-back effect.[3] In that chapter, the CDR is used as a proxy variable to examine the significance of the bounce-back effect and to divide between the expansionary and recession periods for the nonlinear panel data model. The changes of the relationship between the stock return and output growth in the expansionary and recession periods are used to examine whether the relationship would be impacted by the business cycle.

However, no study has touched the topic whether the recession period contains other important information that can be used to examine the relationship between the stock return and economic growth. Domain and Louton [2], Henry and Olekalns [13], and Henry et al. [9] find that the relationship is significantly positive only in the recession period, but the authors do not explain the reasons for this outcome. Beaudry and Koop [14], Henry and Olekalns [13], and Henry et al. [9] argue that in the recession period, there is the bounce-back effect that contributes to the significance of the relationship; however, the authors do not further investigate the bounce-back effect. The empirical study of [2] finds that there is the nonlinear threshold effect in the relationship between the stock return and real economic activities. Henry et al. [9] find that when the economy is in the expansionary stage, the stock return cannot predict the output growth, while in the recession period, the stock return could well predict the output growth.

Does the recession period contain information that would impact the relationship between the stock return and economic growth? The answer can be found in **Figure 1**.[4] During a business cycle, the economy experiences peak, recession, depression, trough, and recovery and then

[1] None study has touched the topic whether the recession period contains other important information that can be used to examine the relationship between the stock return and economic growth.

[2] Most of OECD members are industrial or developed countries.

[3] Friedman [15] after the economy passes the bottom of the business cycle and enters a recovery period, the output will return to the original growth trend, is called the Friedman-type asymmetry. The bounce-back effects are one of the Friedman-type asymmetry.

[4] In **Figure 1**, the CDR and new CDR (NCDR) criteria are used to divide different stages of the business cycle, which is different from the criterion listed in most textbooks.

http://dx.doi.org/10.5772/63629

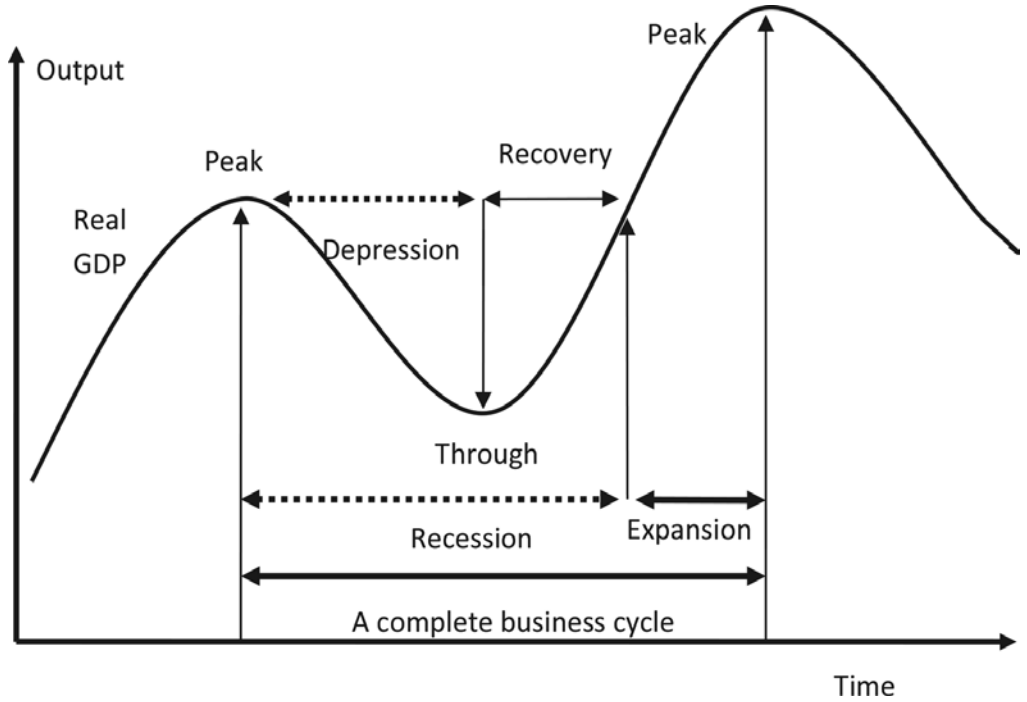

Figure 1. Business cycle process. Note: The CDR and new CDR criteria are adopted to divide the different stages of the business cycle in this figure, which is different from the criterion listed in most textbooks.

moves to another peak and completes a cycle. The recession period actually contains the depression and recovery subperiods. Depression is defined as a period that the economic situation still deteriorates and there is no sign of improvement. Recovery is defined as a period that the economy has already passed the trough; there are signs of improvement, but the economy has not returned to the original growth path. Although both the depression and recovery subperiods are within the recession period, people would have different expectations toward the future in the two different subperiods. In addition, since the stock market always reveals people's expectations of economic growth in advance, the relationship between the stock return and economic growth may differ in the two subperiods.

As to the research method on this field, previous studies often employ the time series data and the linear model; examples include in [16–19]. It is well known that the time series data structure suffers from the following two problems. First, there may not be enough observations in each subperiod, if one divides the sample period into several subperiods. Second, applying individual country data into a multicountry analysis, one may ignore the impact from the economic integration, which may cause the testing power inefficient problem. There are some

problems associated with the use of the cross country data as well, for example, ignoring the impact from the time. To avoid these problems, many researches employ the panel data. This data structure has both the time and the section dimensions; therefore, the empirical result can capture the difference among sample countries and the dynamic changes from time to time. In addition, the two dimensional characteristic also increases the observation numbers, which enhances the degree of freedom.

In the present chapter, the empirical model is modified from the dynamic panel data model (DPDM) of [9]. For the DPDM, if one uses the traditional fixed effect method to estimate the model, it may lead to biased estimation results, because of the correlation between the lagged explained variables and the residual, a problem not addressed in [9]. One way to conquer this problem is to estimate the DPDM with the generalized method of moment (GMM) estimation proposed by Arellano and Bond [20].

The empirical results of this study show that in the recession periods, the stock return significantly impacts the economic growth. In addition, in some of the Asian emerging markets, the stock return is the leading indicator of the economic growth only in the recovery subperiod. As to the developed countries, the stock return is the leading indicator of the economic growth only in the depression subperiod. The empirical results have the following implications and contributions. First, for the international companies, the results can be used to avoid the misunderstandings of the recession process and the relationship between the stock return and economic growth. Second, for the investment organizations and the financial professionals, the results can help them better understand the business cycle and teach the investors the true meaning of the CDR. Third, this study contributes to the filed by further investigate the recession periods with well-established research methods.

This chapter is organized as follows. Section 1 is the introduction. Section 2 discusses and analyzes the data. The research methodology is listed in Section 3. Section 4 shows the empirical results. Section 5 is the conclusion.

2. Data analysis

The data set of this chapter contains quarterly data of stock index, Gross Domestic Product, and consumer price index (CPI) (or Gross Domestic Product deflator) of 26 countries, including the G7 countries (The USA, UK, Canada, France, German, Italy, and Japan), five Asian countries (the Philippines, Singapore, Hong Kong, Korea, and Taiwan), 12 OECD countries (Australia, Austria Belgium, Denmark, Finland, Mexico, the Netherlands, New Zealand, Norway, Spain, Sweden, and Switzerland), Israel, and South Africa. In addition, we divided the sample countries into three groups, one can see the impact of the development level on the relationship between the stock return and economic growth. Group A has five Asian countries; group B has the G7 countries; group C is contains the 12 OECD members. All the data come from the International Financial Statistics (IFS) database of International Monetary Fund (IMF) and the AREMOS database. The longest sample period is from the first quarter of 1982 to the fourth quarter of 2009. Please refer to Appendix A for detailed descriptions of the data.

http://dx.doi.org/10.5772/63629

The two major variables of this chapter are the economic growth rate (*ry*) and the stock return (*rs*):

$$ry_{it} = \Delta \log(GDP_{it}) \times 100, \quad rs_{it} = \Delta \log(ST_{it}) \times 100,$$

where GDP_{it} is the real GDP, ST_{it} is the stock index, ry_{it} is the economic growth rate, and rs_{it} is the stock return; *i* indicates the country and *t* the time; "Δ" denotes the first difference.

3. Research methodology

Beaudry and Koop [14] propose the CDR indicator to capture the stages of business cycle:

$$CDR_{i,t} = \max\{Y_{i,t-s}\}_{s\geq 0}^{t} - Y_{i,t} \geq 0 \tag{1}$$

where $Y_{i,t}$ is the output of period *t*. Eq. (1) tells that the CDR is the output difference between the largest output of period $t - s$ and the output of period *t*.[5] Although the CDR is very sensitive at capturing the recession period, the indicator can only differentiate between the expansionary and recession periods.

Bradley and Jansen [12] argue that the recession period captured by CDR > 0 is a mixed, rather than a pure, recession period; therefore, the authors proposed a new CDR, the NCDR, to capture the pure recession period. Utilizing the output growth rate (ΔY_t), the NCDR divides the mixed recession period into two subperiods, the depression period where the economy is approaching the trough and the recovery period where the economy is approaching the next peak. Bradley and Jansen [12] name the NCDR established the CDR1 > 0 for the depression subperiod and the CDR2 > 0 for the recovery subperiod. The CDR1 and CDR2 indicators are specified as follows:

$$CDR1_{i,t} = \max\{Y_{i,t-s}\}_{s\geq 0}^{t} - Y_{i,t} \geq 0 \quad if\ \Delta Y_{i,t} < 0, \tag{2a}$$

$$CDR2_{i,t} = \max\{Y_{i,t-s}\}_{s\geq 0}^{t} - Y_{i,t} \geq 0 \quad if\ \Delta Y_{i,t} \geq 0. \tag{2b}$$

In the case that $CDR1 = 0$ and $CDR2 = 0$, it indicates that the economy is in the expansionary period, same as the case that CDR = 0 in [14]. Because the NCDR has the benefits of capturing the pure recession period, in the empirical model of this chapter, the NCDR is employed to divide business cycle stages.

[5] If the CDR > 0, that mean the economy is entering the recession period. With the same rationale, if a country is in the expansionary period, then the CDR = 0.

Using the *CDR* as the switched factor to identify the business cycle periods, Henry et al. [9] construct the single-variate nonlinear panel data model:

$$ry_{i,t} = \begin{cases} \alpha_i + \sum_{j=1}^{4} \beta_{1,j} ry_{i,t-i} + \sum_{j=1}^{4} \pi_{1,j} rs_{i,t-j} + \varepsilon_{1i,t} & CDR_{i,t-1} = 0 \\ (\alpha_i + \phi) + \sum_{j=1}^{4} \beta_{2,j} ry_{i,t-j} + \sum_{j=1}^{4} \pi_{2,j} rs_{i,t-j} + \varepsilon_{2i,t} & CDR_{i,t-1} > 0 \end{cases}, \quad (3)$$

where $ry_{i,t}$ denotes the economic growth rate; $rs_{i,t}$ is the stock return; α_i, φ, β_{kj}, and δ_{kj} are coefficients; ε_{kit} is the error term. k = 1, 2. Substituting the CDR of Eq. (3) with the NCDR specified in Eqs. (2a) and (2b) and defining $\Delta Y_{i,t} < 0$ the depression subperiod and $\Delta Y_{i,t} \geq 0$ the recovery subperiod, one can revise the model of Henry et al. [9] as

$$ry_{i,t} = \begin{cases} \alpha_i + \sum_{j=1}^{4} \beta_{1,j} ry_{i,t-i} + \sum_{j=1}^{4} \pi_{1,j} rs_{i,t-j} + \varepsilon_{1i,t} & if\ CDR1_{i,t-2} = CDR2_{i,t-2} = 0 \\ (\alpha_i + \phi) + \sum_{j=1}^{4} \beta_{2,j} ry_{i,t-j} + \begin{cases} (\sum_{j=1}^{4} \delta_{1,j} rs_{i,t-j}) \times DV1 & if\ CDR1_{i,t-2} > 0 \\ (\sum_{j=1}^{4} \delta_{2,j} rs_{i,t-j}) \times DV2 & if\ CDR2_{i,t-2} > 0 \end{cases} + \varepsilon_{2i,t} \end{cases}, \quad (4)$$

where $ry_{i,t}$ denotes the economic growth rate; $rs_{i,t}$ denotes the stock return; α_i, φ, β_{kj}, and δ_{kj} are the coefficients; ε_{kit} is the error term; *CDR*1 and *CDR*2 are the switched factors; DV*k*( ) is the dummy variable, where DV*k* = 1 if the condition inside the parenthesis holds, DV*k* = 0, otherwise. *K* = 1~2. Eq. (4) is the primary empirical model of this chapter.

The readers might be curious why not to construct a three-regime DPDM. The primary reason is that a three-regime DPDM would lead to many differences from the model of [9] to compare the empirical findings. In addition, the three-regime model costs lots of degrees of freedom. Because of these two reasons, what is done here is to derive the depression and recovery subperiods from the recession period, rather than specifying three regimes.

To estimate the nonlinear DPDM, it will be too complicated if one considers the nonlinearity and the dynamic panel data characteristic at the same time in the estimation.[6] A better way to do is to use the two-step method to estimate the nonlinear DPDM. First, the exogenously given switched factors (the CDR or NCDR) are employed to divide the regimes; in the meantime, the dummy variable can reveal the nonlinear relationship between the variables. Second, the

[6] In this paper, the model is a linear panel data model with the characteristic of nonlinearity given by the dummy variables. Therefore, the GMM estimation still can be used to estimate the DPDM and the heteroskedastic residual problem can be avoided.

http://dx.doi.org/10.5772/63629

GMM estimation, named AB-GMM afterward, is utilized to deal with the "dynamic" panel data characteristic that is caused by the inclusion of the lagged dependent variable in the regressors. When one estimates Eq. (4) with AB-GMM estimation, since the variables will be first differenced, the constant term will disappear.

4. Empirical results and analyses

To avoid the spurious regression problem, one should examine whether the panel data are stationary by conducting the unit root test.[7] The result of the panel data unit root test is shown in **Table 1**, shows that the variables are stationary.

	ry		rs	
Method	**value**	**p-value**	**value**	**p-value**
Null: Unit root (assumes common unit root process)				
Levin, Lin & Chu t*	-9.48***	(0.00)	-44.25***	(0.00)
Breitung t-stat	-1.11	(0.13)	-25.93***	(0.00)
Null: Unit root (assumes individual unit root process)				
Im, Pesaran and Shin W-stat	-24.39***	(0.00)	-39.21***	(0.00)
ADF - Fisher Chi-square	631.40***	(0.00)	1072.45***	(0.00)
PP - Fisher Chi-square	949.10***	(0.00)	1069.31***	(0.00)

Notes: The "*ry*" is the economic growth rate, and "*rs*" is the stock return. The above five types of panel unit root tests: Levin et al. [22], Breitung [23], Im et al. [24], Fisher-type tests using ADF and PP tests (Maddala and Wu [25] and Choi [26])."*", "**", and "***" denote the 1% significant level.

Table 1. Panel unit-root test.

At this moment, the CDR (or NCDR) is employed as the switched factor in the nonlinear model to divide between the expansionary and recession periods. The switched factor has been chosen and the switched point has been determined. The lagged period is 4,[8] same as the setting of [9]. The only thing that is adjustable here is the delay period. We find the 1 and 2 periods delay CDR are well, which indicates that it is appropriate to use the delay CDR as the switched factor. In this chapter, the delay period is set to 2 to meet the 5% significant level condition. Because of this, if an economy has a sequence of negative economic growth rates in two seasons, then the economy could be viewed as entering the recession period.[9]

[7] For the spurious regression problem, please refer to Ref. [21].

[8] There is reason to choose four lagged periods. Because the data are quarterly data, so four lagged periods could cover the whole year and capture the seasonal characteristics.

[9] The way to identify whether an economy is entering a recession period is to see whether GDP is decreasing in sequentially two seasons. If it is, then this economy is said to experience recession.

Whole Sample	**(Expansionary period)**			**(Recession period)**	
	$CDR_{i,t-2}$=0			**$CDR_{i,t-2}$>0**	
Variable	**Coefficient**	**p-value**	**Variable**	**Coefficient**	**p-value**
Obs.	**1245**		**Obs.**	**1449**	
A	**0.04**	**(0.87)**	$\alpha + \varphi$	**2.37*** **	**(0.00)**
$ry_{i,t-1}$	-0.92***	(0.00)	$ry_{i,t-1}$	-0.88***	(0.00)
$ry_{i,t-2}$	0.02	(0.67)	$ry_{i,t-2}$	-0.17**	(0.02)
$ry_{i,t-3}$	0.63***	(0.00)	$ry_{i,t-3}$	-0.23**	(0.02)
$ry_{i,t-4}$	-0.25**	(0.03)	$ry_{i,t-4}$	0.03	(0.44)
$rs_{i,t-1}$	0.04**	(0.04)	$rs_{i,t-1}$	0.06**	(0.03)
$rs_{i,t-2}$	0.01	(0.61)	$rs_{i,t-2}$	0.11**	(0.05)
$rs_{i,t-3}$	-0.01	(0.61)	$rs_{i,t-3}$	0.05**	(0.05)
$rs_{i,t-4}$	0.01	(0.41)	$rs_{i,t-4}$	-0.09	(0.23)
	$\sum_{p=1}^{4} \pi_{1p}$ =0.05			$\sum_{p=1}^{4} \pi_{2p}$ =0.13	
Wald Test	H_0: $\pi_{11} = \pi_{12} = \pi_{13} = \pi_{14} = 0$			H_0: $\pi_{21} = \pi_{22} = \pi_{23} = \pi_{24} = 0$	
Chi-square	8.39*		Chi-square	13.38***	
p-value	(0.08)		p-value	(0.01)	

Notes: The "*ry*" is the economic growth rate, and "*rs*" is the stock return; *i* indicates the country and *t* the time. "CDR" is abbreviated from current depth of recession that proposed by [14]. The "Obs" is observation number. "*", "**", and "***" denote the 10%, 5%, and 1% significant levels.

Table 2. Dynamic panel data OLS estimation with CDR switched factor.

After all the parameters have been decided, one can proceed to estimate Eq. (3). For comparison, both the OLS and AB-GMM estimations are performed and the results are listed in **Tables 2** and **3**, respectively. For the coefficient joint test result, in the expansionary period, the stock return cannot explain the economic growth in **Table 3**, which is conflict with the result in **Table 2**. In the recession period, the stock return can significantly explain the economic growth in both **Tables 2** and **3**. Please note that the estimation result in **Table 3** is more consistent with what are found in the literatures (including [9]). In addition, the result in **Table 3** is obtained with the AB-GMM estimation, which can avoid the biased estimation caused by the OLS estimation.[10]

[10] When there are lagged dependent variables in the regressor, the model becomes the DPDM. If one still estimates the model with the OLS estimation, then there would be biased estimation results.

http://dx.doi.org/10.5772/63629

whole sample	**(Expansionary period)**			**(Recession period)**	
	$CDR_{i,t-2}$=0			**$CDR_{i,t-2}$>0**	
Obs.	**1231**		**Obs.**	**1437**	
Variable	**Coefficient**	**p-value**	**Variable**	**Coefficient**	**p-value**
$ry_{i,t-1}$	-0.87***	(0.00)	$ry_{i,t-1}$	-1.23***	(0.00)
$ry_{i,t-2}$	0.30***	(0.00)	$ry_{i,t-2}$	-0.45**	(0.02)
$ry_{i,t-3}$	1.18***	(0.00)	$ry_{i,t-3}$	-0.68***	(0.00)
$ry_{i,t-4}$	0.14	(0.32)	$ry_{i,t-4}$	-0.11	(0.37)
$rs_{i,t-1}$	0.02	(0.14)	$rs_{i,t-1}$	0.04	(0.24)
$rs_{i,t-2}$	0.02	(0.12)	$rs_{i,t-2}$	0.13***	(0.00)
$rs_{i,t-3}$	0.00	(0.97)	$rs_{i,t-3}$	0.12***	(0.00)
$rs_{i,t-4}$	-0.01	(0.50)	$rs_{i,t-4}$	0.02	(0.51)
	$\sum_{p=1}^{4} \pi_{1p} = 0.03$			$\sum_{p=1}^{4} \pi_{2p} = 0.31$	
Wald Test	$H_0: \pi_{11} = \pi_{12} = \pi_{13} = \pi_{14} = 0$			$H_0: \pi_{21} = \pi_{22} = \pi_{23} = \pi_{24} = 0$	
Chi-square	4.89		Chi-square	21.86***	
p-value	(0.30)		p-value	(0.00)	
Instrument rank	104		Instrument rank	204	
J-statistic	110.41		J-statistic	220.20	
p-value	(0.15)		p-value	(0.11)	

Notes: When one estimates Eq. (3) with AB-GMM estimation, since the variables will be first differenced, the constant term will disappear. The "*ry*" is the economic growth rate, and "*rs*" is the stock return; *i* indicates the country and *t* the time. "CDR" is abbreviated from current depth of recession that proposed by [14]. The "Obs" is observation number. Under the null hypothesis that the over-identifying restrictions are valid, the reported *J*-statistic is simply the Sargan statistic, $\chi^2_{p-k'}$ where *k* is the number of estimate coefficients and *p* is the instrument rank. "*", "**", and "***" denote the 5%, and 1% significant levels.

Table 3. Dynamic panel data AB-GMM estimation with CDR switched factor.

The AB-GMM method is used to estimate Eq. (4). **Table 4** reports the full sample estimation result. In the recession period, in the depression or the recovery subperiods, the coefficient joint test result is significant, which indicates there is no difference between the two subperiods and that the stock return can significantly explain the economic growth in two subperiods.

whole sample(Expansionary period)				**(Recession period)**	
	$CDR1_{i,t-2}$=0	**$CDR2_{i,t-2}$=0**		**$CDR1_{i,t-2}$>0**	**$CDR2_{i,t-2}$>0**
Obs.	**1231**		**Obs.**	**1389**	
Variable	Coefficient	p-value	Variable	Coefficient	p-value
$ry_{i,t-1}$	-0.87***	(0.00)	$ry_{i,t-1}$	-1.37***	(0.00)
$ry_{i,t-2}$	0.30***	(0.00)	$ry_{i,t-2}$	0.78**	(0.02)
$ry_{i,t-3}$	1.18***	(0.00)	$ry_{i,t-3}$	0.37	(0.29)
$ry_{i,t-4}$	0.14	(0.32)	$ry_{i,t-4}$	0.28*	(0.06)
				$DV1(CDR1_{t-2}>0)$	
$rs_{i,t-1}$	0.02	(0.14)	$rs_{i,t-1}*DV1$	0.38*	(0.07)
$rs_{i,t-2}$	0.02	(0.12)	$rs_{i,t-2}*DV1$	0.07	(0.52)
$rs_{i,t-3}$	0.00	(0.97)	$rs_{i,t-3}*DV1$	0.16	(0.20)
$rs_{i,t-4}$	-0.01	(0.50)	$rs_{i,t-4}*DV1$	0.28**	(0.02)
				$DV2(CDR2_{t-2}>0)$	
			$rs_{i,t-1}*DV2$	0.00	(0.99)
			$rs_{i,t-2}*DV2$	0.26*	(0.09)
			$rs_{it-3}*DV2$	0.05	(0.81)
			$rs_{i,t-4}*DV2$	-0.34**	(0.03)
	$\sum_{p=1}^{4}\pi_{1p}=0.03$			$\sum_{p=1}^{4}\delta_{1p}=0.89$	$\sum_{p=1}^{4}\delta_{2p}=-0.03$
Wald Test	$H_0: \pi_{11}=\pi_{12}=\pi_{13}=\pi_{14}=0$			$H_0: \delta_{11}=\delta_{12}=\delta_{13}=\delta_{14}=0$	
Chi-square	4.89		Chi-square	10.27**	
p-value	(0.30)		p-value	(0.04)	
				$H_0: \delta_{21}=\delta_{22}=\delta_{23}=\delta_{24}=0$	
			Chi-square		11.19**
			p-value		(0.02)
Instrument rank	104		Instrument rank	112	
J-statistic	110.41		J-statistic	117.85	
p-value	(0.15)		p-value	(0.11)	

Note: The NCDR proposed by Bradley and Jansen [12] is employed as the switched factor in the model.
"*" and "***" denote the 10% and 1% significant levels.

Table 4. Dynamic panel data AB-GMM estimation with NCDR switched factor.

http://dx.doi.org/10.5772/63629

In the following, the estimation method of **Table 4** is repeated on the estimation of groups A to C. The estimation result of group A to C is combined listed in **Table 5** (divide into three part), group A in the upper, group B in the middle, and group C in the lower.

	(Expansionary period)			(Recession period)	
	$CDR1_{i,t-2}=0$	$CDR2_{i,t-2}=0$		$CDR1_{i,t-2}>0$	$CDR2_{i,t-2}>0$
Upper part	**group A**				
	$\sum_{p=1}^{4}\pi_{1p}=0.05$			$\sum_{p=1}^{4}\delta_{1p}=0.15$	$\sum_{p=1}^{4}\delta_{2p}=0.63$
Wald Test	$H_0:\pi_{11}=\pi_{12}=\pi_{13}=\pi_{14}=0$			$H_0:\delta_{11}=\delta_{12}=\delta_{13}=\delta_{14}=0$	
Chi-square	5.19		Chi-square	2.15	
p-value	(0.27)		p-value	(0.71)	
				$H_0:\delta_{21}=\delta_{22}=\delta_{23}=\delta_{24}=0$	
			Chi-square		10.80***
			p-value		(0.03)
Middle part	**group B**				
	$\sum_{p=1}^{4}\pi_{1p}=0.03$			$\sum_{p=1}^{4}\delta_{1p}=0.09$	$\sum_{p=1}^{4}\delta_{2p}=0.04$
Wald Test	$H_0:\pi_{11}=\pi_{12}=\pi_{13}=\pi_{14}=0$			$H_0:\delta_{11}=\delta_{12}=\delta_{13}=\delta_{14}=0$	
Chi-square	1.18		Chi-square	16.37***	
p-value	(0.88)		p-value	(0.00)	
				$H_0:\delta_{21}=\delta_{22}=\delta_{23}=\delta_{24}=0$	
			Chi-square		3.00
			p-value		(0.56)
Lower part	**group C**				
	$\sum_{p=1}^{4}\pi_{1p}=0.23$			$\sum_{p=1}^{4}\delta_{1p}=1.54$	$\sum_{p=1}^{4}\delta_{2p}=0.58$
Wald Test	$H_0:\pi_{11}=\pi_{12}=\pi_{13}=\pi_{14}=0$			$H_0:\delta_{11}=\delta_{12}=\delta_{13}=\delta_{14}=0$	
Chi-square	22.53***		Chi-square	8.66*	
p-value	(0.00)		p-value	(0.07)	
				$H_0:\delta_{21}=\delta_{22}=\delta_{23}=\delta_{24}=0$	
			Chi-square		4.06
			p-value		(0.40)

Note: Table 5 just show the estimation results of three groups partly.
"*" and "***" denote the 10%, and 1% significant levels.

Table 5. Dynamic panel data AB-GMM estimation with NCDR switched factor.

From **Table 5** (upper part), one can see that the coefficients are significantly positive only in the recovery subperiod. This tells that for group A (five Asian emerging countries), the stock return can explain the economic growth only in the recovery period. **Table 5** (middle part) shows that the coefficients are significantly positive only in the depression subperiod, which indicates that for group B (the G7 countries), the stock markets will go down before the economies start to grow.

The economic rationale behind this is as follows. Since the stock return and economic growth are positively correlated in the G7 countries, when they enter the depression subperiod, the stock market would go down to reveal the upcoming depressions. When the G7 countries enter the recovery subperiod, their production and consumption will increase. At this time, the G7 countries will place many orders on the Asian emerging markets, and this will help the Asian emerging markets grow and their firms perform well. These outcomes will be reflected by the stock markets in these emerging markets; with more foreign investments from the developed countries, these stock markets will stay in the bull status for a long time. This is why the stock market can significantly explain the economic growth in the recovery subperiod. Although both the Asian emerging markets and the G7 countries are all in the recovery subperiod, the economies will not grow as fast. Moreover, because developed countries tend to invest in high return foreign stock markets, there is no significant relationship between the stock return and economic growth in the G7 countries in the recovery subperiod.

Table 5 (lower part) reports the estimation result for group C (12 OECD members), the coefficients are significantly positive only in the depression subperiod, same as the result in **Table 5** (middle part). In addition, in the expansionary period, the stock return can significantly explain the economic growth, which is different from the results of other subperiod estimations. The results in **Table 5** (middle and lower parts) can be used to derive the results of **Table 4** that the stock return can explain the economic growth in both the depression and recovery subperiods.

The results of **Tables 5** are not quite the same as the result of **Table 4** (the whole sample estimation), which indicates that one cannot apply the conclusion of **Table 4** to every case. Some of the effects may be "cancelled out" by pooling all the countries into one sample.

The empirical findings of this chapter could benefit the corporations, financial companies, as well as regular investors. The contribution of this chapter can be summarized as follows. First, the empirical findings could avoid corporations from misunderstanding the recession period. In the recession period, the government tends to reduce the interest rate and enhance the government spending to stimulate the economy. When making future operation and finance decisions, if the decision maker can seize the chance to adjust the factory size or to raise corporate debts in the recession period or to finance in the expansionary period, it would be beneficial to the corporation. Second, the findings help financial companies or financial supervisors better understand the business cycle. Macroeconomic analyses and business cycles are crucial factors to investment decisions to maximize capital gains and to minimize risks from market fluctuations. Third, the findings help regular investors better understand the business cycle. The business cycle information can help regular investors with medium or long term investment decisions and avoid capital loss from short term fluctuations in the market.

http://dx.doi.org/10.5772/63629

Fourth, the empirical findings prove that there does exist useful information in the recession period.

5. Conclusions

In this study, the existence of the subperiods of the recession period is observed by examine the business cycle plot of **Figure 1**, this chapter make up this gap by constructing a nonlinear DPDM to investigate the relationship between the stock return and economic growth in the subperiods. The finding of the present chapter can be summarized as follows.

First, the GMM estimation is adopted for the DPDM estimation to avoid possible bias from the OLS estimation. The NCDR proposed by Bradley and Jansen [12] is employed as the switched factor in the model. The empirical result shows that the stock market performance can be a leading indicator for the economic growth, especially in the recession period. This finding is consistent with the findings in previous studies.

Second, the empirical results show that in the whole sample estimation, the stock return can significantly explain the economic growth in the two subperiods of the recession period. In the estimation with different country development levels, it is found that in the Asian emerging markets, the stock return can significantly explain the economic growth only in the recovery period. The reason for this outcome is as follows. Generally speaking, the emerging markets have higher economic growth rates than the most of the developed countries do. When entering the recovery period, the emerging markets can attract more foreign funding into their stock markets. As to the developed countries, the stock return can significantly explain the economic growth only in the depression period. The reason for this result is the following. The developed countries lead the development of the world economy. When the developed countries enter the depression period, the investors will withdraw from the stock markets to avoid the risk of loss, which in turn, causes the stock markets to go down and results in a positive relationship between the depression and the down-turn stock market. This result indicates that various development levels will have different impact on the relationship between the stock return and economic growth. If one would like to use the stock market information to predict the economic growth, one must first ascertain the country's development status.

Appendix A. Data summary (1982Q1–2009Q4)

Country name	Stock market index		GDP	GDP Deflator
Australia	SHARE PRICES:(IMF)	○	○	CPI
Austria	SHARE PRICES (IMF)	○	○	CPI

Country name	Stock market index		GDP	GDP Deflator
Belgium	SHARE PRICES (IMF)	○	○	CPI
Canada	CL.TORONTO STOCK PRICES	○	○	CPI
Denmark	SHARE PRICES(IMF)	89q4~09q4	○	CPI
Finland	SHARE PRICES(IMF)	○	○	CPI
France	SHARE PRICES(IMF)	○	○	CPI
Germany	DAX 30 PERFORMANCE	○	○	GDP deflator
Hong Kong	HANG SENG	○	○	CPI
Israel	SHARE PRICE INDEX (IMF)	○	○	CPI
Italy	SHARE PRICES(IMF)	○	○	CPI
Japan	NIKKEI 225	○	○	CPI
Korea	KOREA SE COMPOSITE	○	○	CPI
Mexico	SHARE PRICES(IMF)	83q1~09q4	○	CPI
Netherlands	SHARE PRICES(IMF)	○	○	CPI
Norway	SHARE PRICES(IMF)	○	○	CPI
New Zealand	SHARE PRICES:(IMF)	○	87q2~09q4	CPI
Philippines	SHARE PRICES:(IMF)	82q1~09q2	○	CPI
Singapore	SINGAPORE STRAITS TIMES	○	○	CPI
South Africa	SHARE PRICES:(IMF)	○	○	CPI
Spain	MADRID SE GENERAL	○	○	CPI
Sweden	SHARE PRICES: (IMF)	○	○	CPI
Switzerland	SWISS MARKET -	88q3~09q4	○	CPI
Taiwan	TAIWAN SE WEIGHTED	○	○	CPI
United Kingdom	SHARE PRICES: (IMF)	○	○	CPI
United States of American	DJAI-30	○	○	CPI

Note: "○" denotes that the corresponding country has the full sample (1982Q1~2009Q4);
THE"CPI" denotes that the consumer price index, the IMF data code is 64.
The IMF data code of SHARE PRICES is 62.
The IMF data code of GDP is 99b.c.
The IMF data code of GDP deflator is 99blr.

http://dx.doi.org/10.5772/63629

Author details

Lee Yuan-Ming[1] and Wang Kuan-Min[2*]

*Address all correspondence to: wkminn@ocu.edu.tw

1 Department of Finance, Southern Taiwan University of Science and Technology, Yongkang District, Tainan, Taiwan

2 Department of Finance, Overseas Chinese University, Taichung, Taiwan

References

[1] Fama, E.F.: Stock return, expected return, real activity. The Journal of Finance. 1990;71:545–546.

[2] Domian, D., Louton, D.: A threshold autoregression analysis of stock returns and real economic activity. International Review of Economics and Finance. 1997;6:167–179.

[3] Hassapis, C.: Financial variables and real activity in Canada. Canadian Journal of Economics. 2003;36:421–442.

[4] Azis, I.J.: Predicting a recovery date from the economic crisis of 2008. Socio-Economic Planning Sciences. 2010;44:122–129.

[5] Choi, J.C., Hauser, S., Kopecky, K.J.: Does stock market predict real activity? time series evidence from the G-7 country. Journal of Banking and Finance.1999;23:1771–1792.

[6] Sarantis, N.: Nonlinearities, cyclical behavior and predictability in stock markets: international evidence. International Journal of Forecasting. 2001;17:459–482.

[7] Aylward, A., Glen, J.: Some international evidence on stock prices as leading of economic activity. Applied Financial Economics. 2000;10:1–14.

[8] Mauro, P.: Stock returns and output growth in emerging and advanced economies. Journal of Development Economics. 2003;71:129–153.

[9] Henry, O.T., Olekalns, N., Thong, J.: Do stock market returns predict changes to output? Evidence from a nonlinear panel data model. Empirical Economics. 2004;29:527–540.

[10] Huang, B.N., Yang, C.W.: Industrial output and stock price revisited: an application of the multivariate indirect causality model. The Manchester School. 2004;72:347–362.

[11] Tsouma, E.: Stock returns and economic activity in mature and emerging markets. The Quarterly Review of Economics and Finance.2009;49:668–685.

[12] Bradley, M.D., Jansen, D.W.: Nonlinear business cycle dynamics: cross-country evidence on the persistence of aggregate shocks. Economic Inquiry. 1997;35:495–509.

[13] Henry, O.T., Olekalns, N.: The effect of recessions on output variability and growth. Southern Economic Journal. 2002;68:683–692.

[14] Beaudry, P., Koop, G.: Do recessions permanently change output? Journal of Monetary Economics. 1993;31:149–163.

[15] Friedman, M.: The 'plucking model' of business fluctuations revised. Economic Inquiry. 1993;31:171–177.

[16] Hamilton, J.D.: A new approach to the economic analysis of nonstationary time series and the business cycle. Econometrica. 1989;57:357–384.

[17] Hamilton, J.D., Lin, G.: Stock market volatility and the business cycle. Journal of Applied Economics. 1996;11:573–593.

[18] Hansen, B.E.: Inference when a nuisance parameter is not identified under the null hypothesis. Econometrica. 1996;64:413–430.

[19] Tsay, R.S.: Testing and modelling multivariate threshold models. Journal of American Statistical Association. 1998;93:1188–1202.

[20] Arellano, M., Bond, S.: Some tests of specification for panel data: Monte Carlo evidence and an application to employment equations. Review of Economic Studies. 1991;58:277–297.

[21] Granger, C.W.J., Newbold, P.: Spurious regressions in economics. Journal of Econometrics. 1974;2:111–120.

[22] Levin, A., Lin, C.F., Chu, C.: Unit root tests in panel data: asymptotic and finite and finite-sample properties. Journal of Econometrics. 2002;108:1–24.

[23] Breitung, J.: The local power of some unit root tests for panel data. In: B. Baltagi (Ed.), Advances in Econometrics 15: Nonstationary Panels, Panel Cointegration, and Dynamic Panel, Amsterdam: JAI Press; 2000. p. 161–178.

[24] Im, K.S., Pesaran, M.H., Shin, Y.: Testing for unit roots in heterogeneous panels. Journal of Econometrics. 2003;115:53–74.

[25] Maddala, G.S., Wu, S.: A comparative study of unit root tests with panel data and a new simple test. Oxford Bulletin of Economics and Statistics. 1999;61:631–652.

[26] Choi, I.: Unit root tests for panel data. Journal of International Money and Finance. 2001;20:249–272.

Edited by Dongbin Lee, Tim Burg and Christos Volos

The book consists mainly of two parts: Chapter 1 - Chapter 7 and Chapter 8 - Chapter 14. Chapter 1 and Chapter 2 treat design techniques based on linearization of nonlinear systems. An analysis of nonlinear system over quantum mechanics is discussed in Chapter 3. Chapter 4 to Chapter 7 are estimation methods using Kalman filtering while solving nonlinear control systems using iterative approach. Optimal approaches are discussed in Chapter 8 with retarded control of nonlinear system in singular situation, and Chapter 9 extends optimal theory to H-infinity control for a nonlinear control system.Chapters 10 and 11 present the control of nonlinear dynamic systems, twin-rotor helicopter and 3D crane system, which are both underactuated, cascaded dynamic systems. Chapter 12 applies controls to antisynchronization/synchronization in the chaotic models based on Lyapunov exponent theorem, and Chapter 13 discusses developed stability analytic approaches in terms of Lyapunov stability. The analysis of economic activities, especially the relationship between stock return and economic growth, is presented in Chapter 14.

ISBN 978-953-51-4173-0

IntechOpen