Modeling, Design, and Simulation of Systems with Uncertainties

Mathematical Engineering

For further volumes:
http://www.springer.com/series/8445

Andreas Rauh • Ekaterina Auer
Editors

Modeling, Design, and Simulation of Systems with Uncertainties

Springer

Editors

Andreas Rauh
University of Rostock
Chair of Mechatronics
Justus-von-Liebig-Weg 6
18059 Rostock
Germany
andreas.rauh@uni-rostock.de

Ekaterina Auer
University of Duisburg-Essen
Faculty of Engineering, INKO
Lotharstr. 63
47057 Duisburg
Germany
auer@inf.uni-due.de

ISBN 978-3-642-15955-8 e-ISBN 978-3-642-15956-5
DOI 10.1007/978-3-642-15956-5
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011929791

Cover design: WMXDesign GmbH, Heidelberg

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

To describe the true behavior of most real-world systems with sufficient accuracy, engineers have to overcome difficulties arising from their lack of knowledge about parts of a process or from the impossibility to characterize it with absolute certainty. For example, measured parameters of (dynamical) systems cannot be determined exactly due to non-negligible equipment imprecision. Other sources of such model inaccuracies are order reduction techniques for complex systems used to simplify the design of their components and corresponding control algorithms. Therefore, both aleatory (due to randomness) and epistemological (due to the lack of knowledge) types of uncertainty have to be taken into account while developing techniques for a model-based analysis or synthesis of systems.

Depending on the application at hand, uncertainties in modeling and measurements can be represented in several different ways. For example, *bounded uncertainties* can be described by intervals, affine forms or general polynomial enclosures such as Taylor models. There are frameworks incorporating corresponding kinds of arithmetics to handle this type of uncertainty, which simultaneously provide verified results. This means that the results are enclosures guaranteed to contain the exact solution sets, assuming that the mathematical models and the corresponding ranges of uncertain quantities are correct.

Another situation arises if the uncertainty can be characterized in the form of probability distributions described, for example, by mean values, standard deviations and higher-order moments (*stochastic uncertainty*). In this case, Bayesian estimation frameworks offer a solution by propagating the corresponding probability density functions. These are handled in terms of either analytic or numeric representations, where the latter approach forms the basis of the well-known Monte Carlo methods.

For both bounded and stochastic uncertainties, there exist specific theoretic concepts and practical applications. The goal of this Special Volume on *Modeling, Design, and Simulation of Systems with Uncertainties* is to make the current research on techniques for uncertainty handling known to a broader circle of researchers and industry representatives. For this purpose, we have collected 16 articles from researchers from Canada, Russia, Germany, USA, France, Austria, Poland, Italy, and

Bulgaria dealing with this topic, from which five were presented at the Minisymposium on *Modeling, Design, and Simulation of Systems with Uncertainties* during the *16th European Conference on Mathematics for Industry ECMI* in Wuppertal, Germany, in July 2010.

The volume is subdivided into two parts. In the first we present works highlighting the theoretic background and current research on algorithmic approaches in the field of uncertainty handling, together with their reliable software implementation. The second part is concerned with real-life application scenarios from various areas including but not limited to mechatronics, robotics, and biomedical engineering.

Rostock, *Andreas Rauh*
Duisburg, *Ekaterina Auer*
March 2011

Acknowledgements

The editors of this Special Volume on *Modeling, Design, and Simulation of Systems with Uncertainties* would like to thank all authors for their high-quality contributions. Moreover, we would like to thank all researchers who supported us by reviewing the individual book chapters and by providing detailed comments and suggestions for improvements. Finally, we would like to thank Prof. Dr. Claus Hillermeier for giving us the opportunity to publish this collection of current research articles in the Springer book series on *Mathematical Engineering*.

Contents

Part I Theoretic Background and Software Implementation

Part II Applications: Uncertainties in Engineering

List of Contributors

Haider Albassam
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: haider.albassam@uni-due.de

Matthias Althoff
Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, USA
e-mail: malthoff@ece.cmu.edu

Felix Antritter
Automatisierungs- und Regelungstechnik, Universität der Bundeswehr München,
Werner-Heisenberg-Weg 39, D-85579 Neubiberg, Germany
e-mail: felix.antritter@unibw.de

Harald Aschemann
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: harald.aschemann@uni-rostock.de

Ekaterina Auer
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: auer@inf.uni-due.de

Neli S. Dimitrova
Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Acad.
G. Bonchev Str. 8, 1113 Sofia, Bulgaria
e-mail: nelid@bio.bas.bg

Serena Doria
Department of Sciences, University G. D'Annunzio, Chieti-Pescara, Italy
e-mail: s.doria@dst.unich.it

Denis Efimov
IMS-lab, University of Bordeaux, 351 cours de la libération, 33405 Talence, France
e-mail: denis.efimov@ims-bordeaux.fr

Darya Filatova
UJK, ul. Krakowska 11, 25-027 Kielce, Poland and Analytical Centre of Russian Academy of Sciences, ul. Vavilova 40, 199911 Moscow, Russia
e-mail: daria_filatova@rambler.ru

Marek Grzywaczewski
Politechnika Radomska, ul. Malczewskiego 20A, 26-600 Radom, Poland
e-mail: mgrzyw@interia.pl

Luc Jaulin
ENSIETA, OSM, Lab-STICC, 2 rue François Verny, 29806 Brest, France
e-mail: jaulinlu@ensieta.fr

Andrés Kecskeméthy
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: andres.kecskemethy@uni-due.de

Michel Kieffer
Laboratoire des Signaux et Systèmes - CNRS - SUPELEC - Univ Paris-Sud, 3 rue Joliot-Curie, F-91192 Gif-sur-Yvette cedex, on leave at LTCI - CNRS - Telecom ParisTech, 46 rue Barault, F-75013 Paris, France
e-mail: michel.kieffer@lss.supelec.fr

Marco Kletting
Multi-Function Airborne Radars (OPES22), Cassidian Electronics, Woerthstr. 85, D-89077 Ulm, Germany
e-mail: marco.kletting@cassidian.com

Georgy V. Kostin
Laboratory of Mechanics of Controlled Systems, Institute for Problems in Mechanics of the Russian Academy of Sciences, Pr. Vernadskogo 101-1, 119526 Moscow, Russia
e-mail: kostin@ipmnet.ru

Mikhail I. Krastanov
Institute of Mathematics and Informatics, Bulg Academy of Sciences
Acad. G. Bonchev Str. 8, 1113 Sofia, Bulgaria
e-mail: krast@math.bas.bg

Bruce H. Krogh
Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, USA
e-mail: krogh@ece.cmu.edu

Wolfram Luther
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: luther@inf.uni-due.de

Mihály Csaba Markót
Fakultät für Mathematik, Universität Wien, Nordbergstr. 15, A-1090 Wien, Austria
e-mail: Mihaly.Markot@univie.ac.at

Mehrdad Moshir
Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr., Pasadena, CA 91109, USA
e-mail: mehrdad.moshir@jpl.nasa.gov

Nedialko S. Nedialkov
Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada, L8S 4K1
e-mail: nedialk@mcmaster.ca

Tarek Raïssi
IMS-lab, University of Bordeaux, 351 cours de la libération, 33405 Talence, France
e-mail: tarek.raissi@ims-bordeaux.fr

Andreas Rauh
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: andreas.rauh@uni-rostock.de

Jöran Ritzke
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Joeran.Ritzke@uni-rostock.de

Vasily V. Saurin
Laboratory of Mechanics and Optimization of Structures, Institute for Problems in Mechanics of the Russian Academy of Sciences, Pr. Vernadskogo 101-1, 119526 Moscow, Russia
e-mail: saurin@ipmnet.ru

Hermann Schichl
Fakultät für Mathematik, Universität Wien, Nordbergstr. 15, A-1090 Wien, Austria
e-mail: Hermann.Schichl@univie.ac.at

Dominik Schindele
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Dominik.Schindele@uni-rostock.de

Sergey P. Shary
Institute of Computational Technologies SB RAS, 6 Lavrentiev ave., 630090 Novosibirsk, Russia
e-mail: shary@ict.nsc.ru

Olaf Stursberg
University of Kassel, Control and System Theory (FB16), Wilhelmshöher Allee 73, D-34121 Kassel, Germany
e-mail: stursberg@uni-kassel.de

Eric Walter
Laboratoire des Signaux et Systèmes - CNRS - SUPELEC - Univ Paris-Sud, 3 rue Joliot-Curie, F-91192 Gif-sur-Yvette cedex, France
e-mail: eric.walter@lss.supelec.fr

Ali Zolghadri
IMS-lab, University of Bordeaux, 351 cours de la libération, 33405 Talence, France
e-mail: ali.zolghadri@ims-bordeaux.fr

Part I
Theoretic Background and Software Implementation

In the first part of this book, we present works highlighting the theoretic background and current research on algorithmic approaches in the field of uncertainty handling together with their reliable software implementation. In Chapter 1, Nedialko S. Nedialkov presents techniques from literate programming which are used in the implementation of the verified ODE solver VNODE-LP. Chapter 2 authored by Sergey P. Shary is concerned with new methods for solving linear systems of equations with interval uncertainties. Andreas Rauh and Harald Aschemann describe techniques for the structural analysis of control and state estimation problems formulated as systems of differential-algebraic equations in Chapter 3. In Chapter 4, Matthias Althoff, Bruce H. Krogh, and Olaf Stursberg consider methods for reachability analysis of linear dynamic processes applicable to high-dimensional system models. A robustness analysis of different tracking control schemes in performed by Marco Kletting and Felix Antritter in Chapter 5. Approaches for set-membership state estimation are presented by Luc Jaulin in Chapter 6, whereas verified global optimization routines for parameter estimation of nonlinear models are discussed by Michel Kieffer, Mihály Csaba Markót, Hermann Schichl, and Eric Walter in Chapter 7. Chapter 8 by Darya Filatova and Marek Grzywaczewski deals with the theory applicable to the design of optimal control strategies for induction heating processes and a robustness evaluation of the obtained results. The first part of this volume is concluded by a contribution on coherent upper and lower conditional previsions authored by Serena Doria.

Chapter 1
Implementing a Rigorous ODE Solver Through Literate Programming

Nedialko S. Nedialkov

Abstract Interval numerical methods produce results that can have the power of a mathematical proof. Although there is a substantial amount of theoretical work on these methods, little has been done to ensure that an implementation of an interval method can be readily verified. However, when claiming rigorous numerical results, it is crucial to ensure that there are no errors in their computation. Furthermore, when such a method is used in a computer assisted proof, it would be desirable to have its implementation published in a form that is convenient for verification by human experts.

We have applied Literate Programming (LP) to produce VNODE-LP, a C++ solver for computing rigorous bounds on the solution of an initial-value problem (IVP) for an ordinary differential equation (ODE). We have found LP well suited for ensuring that an implementation of a numerical algorithm is a correct translation of its underlying theory into a programming language: we can split the theory into small pieces, translate each of them, and keep mathematical expressions and the corresponding code close together in a unified document. Then it can be reviewed and checked for correctness by human experts, similarly to how a scientific work is examined in a peer-review process.

1.1 Introduction

Interval numerical methods produce results that can have the power of a mathematical proof. Typically, such a method computes bounds that are guaranteed to contain the true solution of a problem, proves that a solution does not exists or it indicates

Nedialko S. Nedialkov
Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada, L8S 4K1
e-mail: nedialk@mcmaster.ca

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_1,

that a solution cannot be found. For example, when computes an enclosure on the solution of an IVP in ODEs, an interval solver first proves that there exists a unique solution and then produces bounds that contain it [10]; when solving a nonlinear equation, an interval method can prove that a region does not contain a solution or computes bounds that contain a unique solution to the problem [30]. For an excellent, up-to-date survey of these methods, see [35].

To date, not much has been done to ensure that the implementation of such a method can be readily verified, and the bounds it computes are indeed rigorous. Showing that an implementation is correct is of paramount importance for these methods, as mathematical rigor cannot be claimed, if we miss to include even a single roundoff error in a computation. Furthermore, when interval software is used in a computer-assisted proof, it would be desirable to have the software published in a form that is convenient for inspection and verification by human experts.

The author released in 2001 VNODE [25, 28], Validated Numerical ODE, a C++ package for computing bounds on the solution of an IVP for an ODE. This package is carefully written and tested, and it had shown to be robust and reliable. While one can check the theory behind VNODE (e.g. in [25]), it would be difficult to show that its C++ translation does not contain errors. The same applies to the other packages for computing bounds in IVPs for ODEs: ADIODES [39], COSY [3], and VSPODE [20]. That is, it also would be difficult to establish the correspondence between underlying theory and source code in these packages. A notable exception is AWA [22], where there is a clear "match" between the theory and the program listing in [22]. Another well-documented implementation is the VODESIA package [5], but unfortunately it is not publicly available.

The above solvers have been used to compute rigorous bounds on solutions in IVP ODEs. For example, VNODE had been employed in applications such as rigorous multibody simulations [2], reliable surface intersection [24, 32], robust evaluation of differential geometry properties of a Bezier surface patch [18], computing bounds on eigenvalues [4], parameter and state estimation [12, 34], rigorous shadowing [7, 8], and theoretical computer science [1].

The author had always been concerned about possible errors in the implementation of VNODE. Obviously, if an error is present, then the works that have employed VNODE may contain invalid results. Moreover, how can one establish that the computed bounds are rigorous, and further, how others can be convinced that the implementation and the results are correct? This came as a major concern of the author of [1]: how one can trust the numerical results of VNODE? He needed a rigorous proof that an algebraic expression involving the solution of a highly nonlinear scalar ODE is less than one; otherwise his theorem would not hold. The strongest assurance argument was of the sort "VNODE has been accurate and reliable", but obviously this is not satisfactory. The value of this expression was approximately 0.999... in multiple precision in MAPLE, but it needed to be proved that it was always smaller than 1. With VNODE we showed that the exact value of this expression is always smaller than one, but still, we did not have an unquestionable proof.

This prompted the author to search for ways to show that not only the implementation is correct, but it can also be checked readily by others. Literate Programming

(LP) [16] was found particularly suitable for this purpose. Using LP, we can produce a *verifiable* implementation in the sense that it can be reviewed and examined for correctness, similarly to how a scientific work is reviewed by human experts in a peer-review process. This is in contrast to mechanical software verification, when a proof tool is applied to verify code against given specifications.

We reimplemented VNODE entirely with LP (along with some algorithmic improvements), which resulted in the VNODE-LP solver [27]. This paper gives an overview of VNODE-LP, elaborates on LP, and illustrates the process of employing it for carrying out a verifiable implementation.

Section 1.2 discusses LP. Section 1.3 presents an overview of VNODE-LP. Examples from its implementation, illustrating our approach using LP, are given in Section 1.4. Section 1.5 elaborates on relevant work. Section 1.6 summarizes our experience.

1.2 Literate Programming and VNODE-LP

Literate programming was introduced as a programming methodology by D. Knuth [14, 15]. Its essence can be captured as in [16, pg. 99]: "...instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do", and introducing concepts "...in an order that is best for human understanding, using a mixture of formal and informal methods that reinforce each other."

When developing a literate program, we break down an algorithm into smaller, easy-to-understand parts, and explain, document, and implement each of them in an order that is more natural for human comprehension, versus order that is suitable for compilation. In a literate program, documentation and code are in one source. The program is an interconnected "web" of pieces of code, referred to as *sections* [14,16] or *chunks* [11,37], which can be presented in any sequence. They are assembled into a program ready for compilation in a *tangle* process, which extracts the source code from the LP source. The documentation is "weaved" in a *weave* process, which prepares it for typesetting [16, 17].

We developed VNODE-LP using the CWEB literate programming tool [17] and its `ctangle` and `cweave` utilities. CWEB enables the inclusion of documentation and C++ code in a CWEB source file, which is essentially a LaTeX file with additional statements for dealing with source code.

From a CWEB file, `cweave` generates a LaTeX file; `cweave` takes care of page layout, indentation, suitable fonts, pretty printing of C/C++ code, and generates extensive cross-index information. Originally, CWEB could deal with TeX input only. The LaTeX `cweb` [36] class allows using LaTeX; the `cweb-hy` class [37], an extension of `cweb`, allows structuring of a LaTeX document in chapters, sections, subsections, etc., and also provides automatic generation of hyperlinks, which are convenient for navigation through the code in the resulting, e.g., PDF file.

The ctangle utility extracts the source code and writes C/C++ files. It also includes line information in the generated files so that handling errors when compiling and debugging can be done in terms of CWEB source files, and not the generated C/C++ files. That is, when syntax errors or warnings are encountered, a compiler gives line numbers in web files, and similarly, when runtime errors are detected, a debugger gives line numbers in web files.

Developing a literate program reduces to writing an article or a book: we present the program in an order that follows our thought process and strive to explain our ideas clearly in a document that should be of publishable quality. For each algorithm in [27], we present its theory first, and then translate parts of it, where the division is such that the code in each part is not difficult to inspect. During development, if errors in compilation or execution occur, we can review the manuscript and update accordingly the CWEB files, without looking into the generated program files (they are for compiler consumption). Similarly, when inspecting VNODE-LP, we can work only with the LP document [27].

This article and [27] are created with CWEB and the cweb-hy class. The latter is composed like a book: with a table of contents, list of figures, hierarchical structure of the presentation, index, and bibliography. This document contains everything related to VNODE-LP: user guide, theory, documentation, source code, example, test cases, makefiles, and gnuplot files used for generating the plots in [27]

All the theory of VNODE-LP is included in [27]. Our goal was to have a self-contained, detailed presentation, so a reviewer would need only [27] when evaluating VNODE-LP. Since all the pieces for verifying the theory and implementation are in [27], if their correctness is confirmed by human experts like in a peer-review process, we may trust, or at least have high-confidence, in the correctness of the implementation of VNODE-LP and accept the bounds it computes as rigorous. When claiming rigor, however, we presume that the operating system, compiler, and the packages VNODE-LP uses do not contain errors affecting its execution.

1.3 Overview of VNODE-LP

We introduce interval arithmetic (IA), state the IVP that is the subject of this work (§1.3.1), and discuss briefly the methods in VNODE-LP and the packages it uses (§1.3.2).

1.3.1 The IVP VNODE-LP *Solves*

The VNODE-LP software builds on IA as defined below. Denote the set of closed (finite, nonempty) intervals on $\mathbb{R}$ by

$$\mathbb{IR} = \left\{ \boldsymbol{a} = [\underline{a}, \overline{a}] \mid \underline{a} \le x \le \overline{a},\ \underline{a}, \overline{a} \in \mathbb{R} \right\}.$$

If $\boldsymbol{a}$ and $\boldsymbol{b} \in \mathbb{IR}$ and $\bullet \in \{+,-,\times,/\}$, then the IA operations are defined as

$$\boldsymbol{a} \bullet \boldsymbol{b} = \{x \bullet y \mid x \in \boldsymbol{a},\ y \in \boldsymbol{b}\},$$

where division is undefined if $0 \in \boldsymbol{b}$.

Now consider the IVP

$$y'(t) = f(t,y), \quad y(t_0) = y_0, \qquad t \in \mathbb{R},\ y \in \mathbb{R}^n. \tag{1.1}$$

where $f : \mathbb{R} \times \mathbb{R}^n$ is sufficiently smooth. As a consequence, the code list of f should not contain for example branches, abs, or min. For more details see [10, 25–29].

Denote the set of n-dimensional interval vectors by $\mathbb{IR}^n$. Given $\boldsymbol{y}_0 \in \mathbb{IR}^n$ and $t_{\text{end}} \neq t_0$ ($t_{\text{end}} \in \mathbb{R}$), VNODE-LP tries to compute a $\boldsymbol{y}_{\text{end}} \in \mathbb{IR}^n$ at t_{end} that contains the solution to (1.1) at t_{end} for all $y_0 \in \boldsymbol{y}_0$. If VNODE-LP cannot reach t_{end}, for example the bounds on the solution become too wide, bounds at some t^* between t_0 and t_{end} are returned.

1.3.2 Methods and Packages

Denote by $y(t_j;t_0,y_0)$ the solution to (1.1) with an initial condition y_0 at t_0, and denote by $\boldsymbol{y}_j$ an enclosure of the solution at t_j. That is,

$$y(t_j;t_0,y_0) \in \boldsymbol{y}_j \quad \text{for all } y_0 \in \boldsymbol{y}_0.$$

This solver proceeds in a one-step manner from t_0 to t_{end}, where it computes bounds at (adaptively) selected points $t_j \in (t_0,t_{\text{end}}]$. On a step from t_j to t_{j+1}, VNODE-LP computes first a priori bounds $\widetilde{\boldsymbol{y}}_j$ such that

$$y(t;t_j,y_j) \in \widetilde{\boldsymbol{y}}_j \quad \text{for all } t \in [t_j,t_{j+1}] \quad \text{and all } y_j \in \boldsymbol{y}_j.$$

Then it finds tight bounds $\boldsymbol{y}_{j+1}$ at t_{j+1} such that

$$y(t_{j+1};t_0,y_0) \in \boldsymbol{y}_{j+1} \quad \text{for all } y_0 \in \boldsymbol{y}_0;$$

see Figure 1.1. To compute these bounds, we use IA, Taylor series expansion of the solution to (1.1) at each integration point, and various interval techniques.

VNODE-LP is based on Taylor series and the Hermite-Obreschkoff [25] methods. It is a fixed-order, variable-stepsize solver. The stepsize is varied such that an estimate of the *local excess* per unit step is below a user-specified tolerance. Typical values for the order (for efficient integration) can be between 20 and 30 [26]; the default order is set to 20.

Generally, VNODE-LP is suitable for computing bounds on the solution of an IVP ODE with point initial conditions or interval initial conditions with a sufficiently small width. If the initial condition set is not small enough and/or long time integration is desired, the COSY package [3] of Berz and Makino can produce tighter

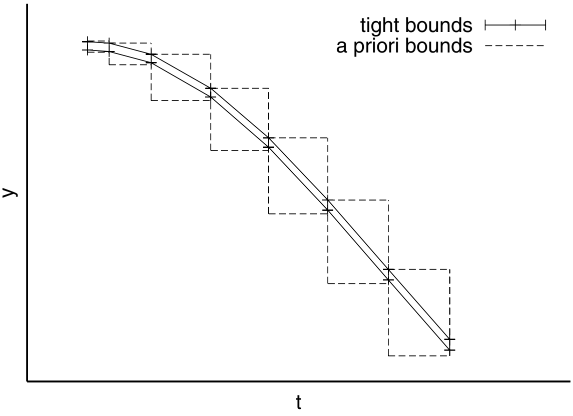


Fig. 1.1: A priori and tight bounds. For this visualization, the tight bounds are connected with lines, which do not necessarily enclose the true solution

bounds than VNODE-LP. Alternatively, one can subdivide the initial interval vector (box) $\boldsymbol{y}_0$ into smaller boxes, perform integrations with them as initial conditions, and build an enclosure of the solution at the desired t_{end}.

We tried to avoid advanced C++ constructs and tried to minimize the dependence of VNODE-LP on the IA package. The present distribution of VNODE-LP compiles with either of the IA packages PROFIL/BIAS [13] or FILIB++ [19]. Recently, the IA package GAOL [6] was used as the IA package in VNODE-LP [9].

The interface to an IA package is encapsulated in 26 small (most of them single line), inline wrapper functions that call functions from it. We aimed at keeping this interface as small as possible, such that another IA package can be incorporated easily by implementing these wrapper functions. For this reason, we do not use, for example, the matrix and vector classes of PROFIL/BIAS, but implement our own matrix and vector operations through the C++ standard template library.

A major component of our solver is the tool for generating Taylor coefficients and Jacobians of Taylor coefficients through automatic differentiation (AD). This is done using the FADBAD++ [40] AD package. We also use LAPACK and BLAS for computing an approximate matrix inverse, which is needed for enclosing the inverse of an interval matrix.

1.4 Examples from VNODE-LP

We illustrate typical steps when developing VNODE-LP: we give examples of two simple functions (§1.4.1) and an example of translating an expression that is part of a function (§1.4.2). We also present a simple program for integrating the Lorenz system (§1.4.3).

1.4.1 Computing h such that $[0,h]\,\mathbf{a} \subseteq \mathbf{b}$

The following problem is from the VNODE-LP implementation: given finite machine intervals $\boldsymbol{a}$ and $\boldsymbol{b}$, where $0 \in \boldsymbol{b}$, find the largest $h \geq 0$ such that $[0,h]\,\boldsymbol{a} \subseteq \boldsymbol{b}$. Here, for $\boldsymbol{x},\boldsymbol{y} \in \mathbb{IR}$, $\boldsymbol{x} \subseteq \boldsymbol{y}$ iff $\underline{x} \geq \underline{y}$ and $\overline{x} \leq \overline{y}$.

We derive a formula for h and then produce the C++ code. By $\triangledown(x/y)$, we denote the rounded towards $-\infty$ result of x/y.

1. If $\underline{a} = \overline{a} = 0$, then $[0,h]\,\boldsymbol{a} = [0,0] \subseteq \boldsymbol{b}$ for any h, and we set $h = \textbf{numeric\_limits}\langle\textbf{double}\rangle{::}\mathit{max}(\,)$, the largest double precision number. Below we assume $\boldsymbol{a} \neq [0,0]$.
2. If $\underline{a} \geq 0$, then $\overline{a} > 0$ and $[0,h]\,\boldsymbol{a} = [0,h\overline{a}] \subseteq [\underline{b},\overline{b}]$ when $h \leq \overline{b}/\overline{a}$. We set $h = \triangledown(\overline{b}/\overline{a})$.
3. If $\overline{a} \leq 0$, then $\underline{a} < 0$ and $[0,h]\,\boldsymbol{a} = [h\underline{a},0] \subseteq [\underline{b},\overline{b}]$ when $h \leq \underline{b}/\underline{a}$. We set $h = \triangledown(\underline{b}/\underline{a})$.
4. If $\underline{a} < 0 < \overline{a}$, then $[0,h]\,\boldsymbol{a} = [h\underline{a},h\overline{a}] \subseteq [\underline{b},\overline{b}]$ when $h = \min\{\underline{b}/\underline{a},\overline{b}/\overline{a}\}$. We set $h = \min\bigl\{\triangledown(\underline{b}/\underline{a}),\ \triangledown(\overline{b}/\overline{a})\bigr\}$.

We translate the above cases into:

1 $\langle h$ such that $[0,h]\boldsymbol{a} \subseteq \boldsymbol{b}$ (intervals) 1$\rangle \equiv$

```
#include <climits>
  using namespace std;
  using namespace v_bias;
  inline double compH(const interval &a, const interval &b)
  {      /* inf(a) returns a̲; sup(a) returns a̅ */
    if (inf(a) ≡ 0 ∧ sup(a) ≡ 0)  return numeric_limits⟨double⟩::max();
    round_down();      /* set rounding mode to −∞ */
    if (inf(a) ≥ 0)  return sup(b)/sup(a);
    if (sup(a) ≤ 0)  return inf(b)/inf(a);
    return std::min(inf(b)/inf(a), sup(b)/sup(a));
  }
```

This code is used in chunk 2

This is a *chunk* of code. It is identified by its name, here "h such that $[0,h]\boldsymbol{a} \subseteq \boldsymbol{b}$ (intervals)". The ctangle program, when extracting the code, orders the chunks based on their names. Each chunk is numbered by cweave, and these numbers are convenient for referencing them in the LP document.

A nice feature of cweave is that it typesets the code in a very readable form, while the code that is typed in a web file does not even need to be indented. Mathematics can be included in a LaTeX form as a comment, and **if** conditions are typeset more like math, rather than C++.

Now, given interval vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, with each component of $\boldsymbol{b}$ containing 0, we wish to find the largest representable $h \geq 0$ such that $[0,h]\boldsymbol{a} \subseteq \boldsymbol{b}$. We write

2 $\langle h$ such that $[0,h]\boldsymbol{a} \subseteq \boldsymbol{b}$ (interval vectors) 2$\rangle \equiv$

```
  ⟨h such that [0,h]a ⊆ b (intervals) 1⟩
```

```
double compH(const iVector&a, const iVector&b)
{
  double hmin = compH(a[0], b[0]);
  for (unsigned int i = 1; i < sizeV(a); i++) {
    double h = compH(a[i], b[i]);
    if (h < hmin) hmin = h;
  }
  return hmin;
}
```

This chunk includes the previous one and calls *compH* on each two components to find h.

1.4.2 Translating Expressions

A method in VNODE-LP can be broken down into a sequence of formulas, and each formula must be implemented carefully, to ensure that all truncation and roundoff errors in a computation are included in the resulting bounds. To achieve this, each formula (or a few formulas) is translated into a chunk. The resulting chunks are put together by `ctangle`, thus obtaining an implementation of the complete method.

Here is another simple example from VNODE-LP's implementation. When propagating bounds on the global excess [25, 27], we need to evaluate

$$\boldsymbol{r}_{j+1} = (A_{j+1}^{-1}\boldsymbol{A}_{j+1})\boldsymbol{r}_j + A_{j+1}^{-1}\boldsymbol{v}_{j+1},$$

where $\boldsymbol{r}_j$ and $\boldsymbol{v}_{j+1}$ are interval vectors, $\boldsymbol{A}_{j+1}$ is an interval matrix, and A_{j+1} is a nonsingular point matrix. The chunk implementing this formula (we omit the declarations of objects and variables) is:

3 $\langle \boldsymbol{r}_{j+1} = (A_{j+1}^{-1}\boldsymbol{A}_{j+1})\boldsymbol{r}_j + A_{j+1}^{-1}\boldsymbol{v}_{j+1}\ 3 \rangle \equiv$ /*

$$\begin{array}{r@{\,}l} \textit{trial\_solution}\to A &= A_{j+1} \\ A &\supseteq \boldsymbol{A}_{j+1} \\ v &\supseteq \boldsymbol{v}_{j+1} \\ \textit{solution}\to r &\supseteq \boldsymbol{r}_j \\ \hline \textit{Ainv} &\ni A_{j+1}^{-1} \text{ if } \textit{ok} \\ \textit{temp} &\supseteq A_{j+1}^{-1}\boldsymbol{v}_{j+1} \\ M &\supseteq A_{j+1}^{-1}\boldsymbol{A}_{j+1} \\ \textit{trial\_solution}\to r &\supseteq (A_{j+1}^{-1}\boldsymbol{A}_{j+1})\boldsymbol{r}_j \\ \textit{trial\_solution}\to r &\supseteq \boldsymbol{r}_{j+1} = (A_{j+1}^{-1}\boldsymbol{A}_{j+1})\boldsymbol{r}_j + A_{j+1}^{-1}\boldsymbol{v}_{j+1} \end{array}$$

*/

```
bool ok = matrix_inverse→encloseMatrixInverse(Ainv, trial_solution→A);
if (ok) {
  multMiVi(temp, Ainv, v);
  multMiMi(M, Ainv, A);
  multMiVi(trial_solution→r, M, solution→r);
  addViVi(trial_solution→r, temp);
}
```

In the comment above the horizontal line, we state informally where the vectors and matrices are stored before executing the code: *trial_solution→A* stores[12] A_{j+1}, v contains $\boldsymbol{v}_{j+1}$, A contains $\boldsymbol{A}_{j+1}$, and *solution→r* contains $\boldsymbol{r}_j$. After the horizontal line, we state each step of the computation, so we can easily check the code that follows against it.

The *encloseMatrixInverse* function computes an interval matrix, output argument *Ainv*, which encloses A_{j+1}^{-1}. If this function computes an enclosure (A_{j+1} is nonsingular and not badly conditioned), then we evaluate the expression. Here *Mi* and *Vi* stand for interval matrix and interval vector, respectively. Obviously, it is not difficult to establish the validity of this code.

Remark 1.1. One may find the explanations here and in [27] containing too much detail. However, our goal is to provide as much detail as possible such that one can readily verify all the steps when going from theory to code.

Remark 1.2. For better understanding, the author has found it helpful to write in comments what is computed, in addition to the exposition before a chunk. We could comment separate lines of code, but it becomes less readable.

1.4.3 Integrating the Lorenz System

We give an example illustrating basic integration with VNODE-LP and showing in more detail how LP works. More examples are given in [27].

With VNODE-LP, the user has to specify first the right side of an ODE problem and then provide a main program. An ODE must be given by a template function for evaluating $y' = f(t, y)$ of the form

```
4 ⟨template ODE function 4⟩ ≡
  template⟨typename var_type⟩
  void ODEName(int n, var_type *yp, const var_type *y, var_type t,
          void *param)
  {
          /* body */
  }
```

[1] For readers not familiar with C++, the operator→ selects a field in a structure when a pointer is being used.

[2] Since *trial_solution→A*, *A*, *v*, *trial_solution→r*, *Ainv*, *temp*, and *M* are C++ objects, they do not appear in bold font, as they are typeset by `cweave` as code.

Here n is the size of the problem, t is the time variable, y is a pointer to input variables, *yp* is a pointer to output variables, and *param* is a pointer to additional parameters that can be passed to this function.

Consider the Lorenz system

$$\begin{aligned} y_1' &= \sigma(y_2 - y_1) \\ y_2' &= y_1(\rho - y_3) - y_2 \\ y_3' &= y_1 y_2 - \beta y_3, \end{aligned}$$

where σ, ρ, and β are constants. This system is encoded in the *Lorenz* function below. The constants have values $\sigma = 10$, $\beta = 8/3$, and $\rho = 28$. We initialize *beta* with the interval containing $8/3$: **interval**(8.0) creates an interval with endpoints 8.0, and **interval**$(8.0)/3.0$ is the interval containing $8/3$.

5 ⟨Lorenz 5⟩ ≡

```
template⟨typename var_type⟩
void Lorenz(int n, var_type *yp, const var_type *y, var_type t,
        void *param)
{
  interval sigma(10.0), rho(28.0);
  interval beta = interval(8.0)/3.0;
  yp[0] = sigma * (y[1] − y[0]);
  yp[1] = y[0] * (rho − y[2]) − y[1];
  yp[2] = y[0] * y[1] − beta * y[2];
}
```

This code is used in chunk 6

We give a simple main program and develop its parts.

6 ⟨simple main program 6⟩ ≡

```
⟨Lorenz 5⟩
int main()
{
  ⟨set initial condition and endpoint 7⟩
  ⟨create AD object 8⟩
  ⟨create a solver 9⟩
  ⟨integrate (basic) 10⟩
  ⟨check if success 11⟩
  ⟨output results 12⟩
  return 0;
}
```

This code is used in chunk 13

The initial condition and endpoint are represented as intervals in VNODE-LP. In this example, they are all point values stored as intervals. The components of *iVector* (interval vector) are accessed like a C/C++ array is accessed.

7 ⟨set initial condition and endpoint 7⟩ ≡

```
  const int n = 3;
  interval t = 0.0, tend = 20.0;

  iVector y(n);
  y[0] = 15.0;
  y[1] = 15.0;
  y[2] = 36.0;
```

This code is used in chunk 6

Then we create an AD object of class FADBAD_AD. It is instantiated with data types for computing Taylor coefficients (TCs) of the ODE solution and TCs of the solution to its variational equation, respectively [25]. To compute these coefficients, we employ the FADBAD++ package [40]. The first parameter in the constructor of FADBAD_AD is the size of the problem. The second and third parameters are the name of the template function.

8 ⟨create AD object 8⟩ ≡

```
  AD ∗ad = new FADBAD_AD(n,Lorenz,Lorenz);
```

This code is used in chunk 6

Now, we create a solver:

9 ⟨create a solver 9⟩ ≡

```
  VNODE ∗Solver = new VNODE(ad);
```

This code is used in chunk 6

The integration is carried out by the *integrate* function. It attempts to compute bounds on the solution at *tend*. When *integrate* returns, either $t = tend$ or $t \neq tend$. In both cases, y contains the ODE solution at t.

10 ⟨integrate (basic) 10⟩ ≡

```
  Solver→integrate(t,y,tend);
```

This code is used in chunk 6

We check if an integration is successful by calling *Solver→successful*( ):

11 ⟨check if success 11⟩ ≡

```
  if (¬Solver→successful())
        cout ≪ "VNODE-LP␣could␣not␣reach␣t␣=␣" ≪ tend ≪ endl;
```

This code is used in chunk 6

Finally, we report the computed enclosure of the solution at t by

12 ⟨output results 12⟩ ≡

```
  cout ≪ "Solution␣enclosure␣at␣t␣=␣" ≪ t ≪ endl;
  printVector(y);
```

This code is used in chunk 6

The VNODE-LP package is in the namespace *vnodelp*. The interface to VNODE-LP is stored in the file `vnode.h`, which must be included in any file using VNODE-LP. We store our program in the file `basic.cc`.

13 ⟨basic.cc 13⟩ ≡

```
#include <ostream>
#include "vnode.h"
  using namespace std;
  using namespace vnodelp;
  ⟨simple main program 6⟩
```

When compiled and executed, the output of this program is

```
Solution enclosure at t = [20,20]
14.30[38161600956570,44725513004334]
9.5[785946141093152,801346480733898]
39.038[2374138960486,4119183796657]
```

It is interpreted as

$$y(20) \in \begin{pmatrix} [14.3038161600956570, 14.3044725513004334] \\ [\ \ 9.5785946141093152,\ \ 9.5801346480733898] \\ [39.0382374138960486, 39.0384119183796657] \end{pmatrix}. \qquad (1.2)$$

These results are produced using PROFIL/BIAS, and the output format is due to the output format of this package. (The platform is x86 Linux with the GCC compiler.) For comparison, if we integrate the Lorenz system with MAPLE using `dsolve` with options `method=taylorseries and abserr=Float(1,-18)`, and with `Digits := 20`, we obtain

$$y(20) \approx \begin{pmatrix} 14.304146251277895001 \\ 9.5793690774871976695 \\ 39.038325167739731729 \end{pmatrix},$$

which is contained in the bounds (1.2).

Needless to say, one can write application programs without LP. In Figure 1.2, we show the code of the above example written in “plain” C++.

Remark 1.3. Here, the chunks are presented in a consecutive order, but as mentioned earlier, they can be in any order.

1.5 Relevant Work

A comprehensive collection of resources on LP, including extensive bibliography is [21]; annotated bibliography of LP until 1991 is [38]. To the best of the author’s knowledge, VNODE-LP is the first LP implementation of an interval package, and the only other implementation of non-trivial *numerical* software appears to be [33].

```
#include <ostream>
#include ”vnode.h”
using namespace std;
using namespace vnodelp;

template<typename var_type>
void Lorenz(int n,var_type*yp, const var_type*y,
            var_type t, void*param) {
  interval sigma(10.0), rho(28.0);
  interval beta = interval(8.0)/3.0;

  yp[0] = sigma*(y[1]-y[0]);
  yp[1] = y[0]*(rho-y[2])-y[1];
  yp[2] = y[0]*y[1]-beta*y[2];
}

int main() {
  const int n = 3;
  interval t = 0.0,tend = 20.0;
  iVector y(n);
  y[0] = 15.0;
  y[1] = 15.0;
  y[2] = 36.0;
  AD *ad= new FADBAD_AD(n,Lorenz,Lorenz);
  VNODE *Solver= new VNODE(ad);
  Solver->integrate(t,y,tend);
  if(!Solver->successful())
    cout<<”VNODE-LP could not reach t = ”<<tend<<endl;
  cout<<”Solution enclosure at t = ”<<t<<endl;
  printVector(y);
  return 0;
  }
```

Fig. 1.2: “Plain” C++ code for the Lorenz example

In [23], LP is used to facilitate the verification of a network security device. The authors propose in [23] that LP techniques are used to “document the entire assurance argument.” According to their experience, rigorous arguments, including machine-generated proofs of theory and implementation, “did not significantly improve the certifier’s confidence” in their validity. One of the main reasons is that specifications and proofs were documented in a manner to facilitate acceptance by mechanical tools rather than humans. Essentially, the authors conclude that LP greatly facilitates the development of assurance arguments that would be more naturally understood by (human) certifiers than descriptions of machine-generated proofs.

A notable methodology for inspecting an implementation is the program function tables approach of D. Parnas [31]. Before considering LP, the author assessed this approach for inspecting VNODE. However, program function tables are suitable when the relation between input and output arguments is represented by a relatively simple function, which is hardly the case with VNODE.

1.6 Summary of Experience

Developing a non-trivial literate program can be time consuming, which manifests itself into a substantial "up-front" investment of time: we focus on writing a high-quality, well-structured, and easy-to-understand document. This requires paying attention to detail and ensuring that no errors are present. Since this process is inherently slow, one is "forced" to write code carefully, reducing the likelihood of errors.

Once the effort is put into writing a good LP document, then little time goes into debugging and testing—instead of trying to discover errors through them, we simply proofread the LP document. Moreover, theory and code can be cross checked against each other, and error in one may be revealed in the other. In addition, since documentation and code are in one source, they can be naturally kept in sync.

In the author's opinion, if one shows that (a) the theory of a method is correct and (b) its implementation is a provably correct translation of the theory, then minimal testing is required. From the author's experience, if he had implemented the original VNODE solver through LP, then less time would have been spent on checking the implementation, debugging, and testing. More importantly, the confidence in the implementation would have been much higher.

There are 14 tests in the distribution of VNODE-LP. Their main purpose is to ensure that the IA package and VNODE-LP are installed properly. Indeed, the few problems reported to the author about VNODE-LP not being able to execute a test successfully were all related to problems in the installation of the underlying IA package.

It does not appear appropriate to use LP at early stages of program development, when prototyping and experimenting with algorithms, design, and interfaces. When a design is settled, and no major changes are anticipated, then one can "cement" the implementation with LP. In our case, VNODE was in a stable state, and no experimenting was needed before investing into VNODE-LP.

The number of C/C++ lines (without comments) in VNODE-LP is 2,030. This is not a large package, but complex "per line of code." The LP document [27] is 218 pages. For much larger programs, LP may not be an attractive option, especially when a software product must be delivered on time. In academia, researchers rarely go beyond prototype, research codes and releasing software packages, let alone devoting a substantial amount of time into producing a book-like manuscript (which may not count as a publication). At least for the above two reasons, LP is not ubiquitous, even though it has existed for more than 25 years.

Although LP may appear prohibitively time consuming, the author believes that the cumulative effort for producing and maintaining a complex program is smaller using LP compared to "traditional" program development. The author also believes that establishing program correctness by reviewing a literate program may be more effective than employing a software verification tool. It requires not only that a proof mechanism is constructed, but also that the corresponding theory, documentation, and software are checked—we may as well inspect the original program.

Finally, an interval method, which is theoretically guaranteed to produce rigorous results, should be implemented and documented with the same rigor as its theory is derived. Guaranteeing rigor due to theory and not of its implementation diminishes the purpose of such a method.

Acknowledgements The author thanks George Corliss, Math Pharr, John Pryce, Spencer Smith, Bingzhou Zheng, and the anonymous referees for careful reading of this article and providing valuable comments.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

References

1. Achlioptas, D.: Setting 2 variables at a time yields a new lower bound for random 3-SAT. Tech. Rep. MSR-TR-99-96, Microsoft Research, Microsoft Corp., One Microsoft Way, Redmond, WA 98052 (1999)
2. Auer, E., Kecskeméthy, A., Tändl, M., Traczinski, H.: Interval algorithms in modelling of multibody systems. In: Numerical Software with Result Verification, *LNCS*, vol. 2991, pp. 132–159. Springer-Verlag (2004)
3. Berz, M.: COSY INFINITY version 8 reference manual. Technical Report MSUCL–1088, National Superconducting Cyclotron Lab., Michigan State University, East Lansing, Mich. (1997)
4. Brown, B.M., Langer, M., Marletta, M., Tretter, C., Wagenhofer, M.: Eigenvalue bounds for the singular Sturm-Liouville problem with a complex potential. J. Phys. A: Math. Gen. **36**(13), 3773–3787 (2003)
5. Dietrich, S.: Adaptive verifizierte lösung gewöhnlicher differentialgleichungen. Ph.D. thesis, University of Karlsruhe, Karlsruhe, Germany (2003)
6. Goualard, F.: Gaol: Not just another interval library (version 2.0.2). http://sourceforge.net/projects/gaol/ (2006)
7. Hayes, W.: Rigorous shadowing of numerical solutions of ordinary differential equations by containment. Ph.D. thesis, Department of Computer Science, University of Toronto, Toronto, Canada (2001)
8. Hayes, W., Jackson, K.R.: Rigorous shadowing of numerical solutions of ordinary differential equations by containment. SIAM J. Numer. Anal. **42**(5), 1948–1973 (2003)
9. Ishii, D.: Simulation and verification of hybrid systems based on interval analysis and constraint programming. Ph.D. thesis, Graduate School of Science and Engineering, Waseda University, Japan (2010)
10. Jackson, K.R., Nedialkov, N.S.: Some recent advances in validated methods for IVPs for ODEs. Appl. Numer. Math. **42**, 269–284 (2002)
11. Johnson, A., Johnson, B.: Literate programming using `noweb`. Linux J., Article No 1 (1997)
12. Kieffer, M., Walter, E.: Nonlinear parameter and state estimation for cooperative systems in a bounded-error context. In: Numerical Software with Result Verification, *LNCS*, vol. 2991, pp. 107–123. Springer-Verlag (2004)
13. Knüppel, O.: PROFIL/BIAS – a fast interval library. Computing **53**(3–4), 277–287 (1994)

14. Knuth, D.E.: The WEB system of structured documentation. Stanford Computer Science Report CS980, Stanford University, Stanford, CA (1983)
15. Knuth, D.E.: Literate programming. Technical report STAN-CS-83-981, Stanford University, Department of Computer Science (1983)
16. Knuth, D.E.: Literate Programming. Center for the Study of Language and Information, Stanford, CA, USA (1992)
17. Knuth, D.E., Levy, S.: The CWEB System of Structured Documentation. Addison-Wesley, Reading, Massachusetts (1993)
18. Lee, C.K.: Robust evaluation of differential geometry properties using interval arithmetic techniques. Master's thesis, Massachusetts Institute of Technology, Department of Ocean Engineering (2005)
19. Lerch, M., Tischler, G., Gudenberg, J.W.V., Hofschuster, W., Krämer, W.: FILIB++, a fast interval library supporting containment computations. ACM Trans. Math. Softw. **32**(2), 299–324 (2006)
20. Lin, Y., Stadtherr, M.A.: Validated solution of initial value problems for ODEs with interval parameters. In: R.L. Muhanna, R.L. Mullen (eds.) Proceedings of 2nd NSF Workshop on Reliable Engineering Computing. Savannah, GA (2006)
21. Literate programming web site. http://www.literateprogramming.com
22. Lohner, R.J.: Einschließung der Lösung gewöhnlicher Anfangs– und Randwertaufgaben und Anwendungen. Ph.D. thesis, Universität Karlsruhe (1988)
23. Moore, A.P., Payne, C.N., Jr.: Increasing assurance with literate programming techniques. In: Proceedings of 11th Annual Conference on Computer Assurance. COMPASS '96, pp. 187–198 (1996)
24. Mukundan, H., Ko, K.H., Maekawa, T., Sakkalis, T., Patrikalakis, N.M.: Tracing surface intersections with a validated ODE system solver. In: G. Elber, G. Taubin (eds.) Proceedings of the Ninth EG/ACM Symposium on Solid Modeling and Applications. Eurographics Press, June 2004 (2004)
25. Nedialkov, N.S.: Computing rigorous bounds on the solution of an initial value problem for an ordinary differential equation. Ph.D. thesis, Department of Computer Science, University of Toronto, Toronto, Canada, M5S 3G4 (1999)
26. Nedialkov, N.S.: Interval tools for ODEs and DAEs. In: SCAN Conference Proceedings, http://www.computer.org/portal/web/csdl/doi/10.1109/SCAN.2006.28 (2006)
27. Nedialkov, N.S.: VNODE-LP — a validated solver for initial value problems in ordinary differential equations. Tech. Rep. CAS-06-06-NN, Department of Computing and Software, McMaster University, Hamilton, Canada, L8S 4K1 (2006). VNODE-LP is available at http://www.cas.mcmaster.ca/~nedialk/vnodelp
28. Nedialkov, N.S., Jackson, K.R.: The design and implementation of a validated object-oriented solver for IVPs for ODEs. Tech. Rep. 6, Software Quality Research Laboratory, Department of Computing and Software, McMaster University, Hamilton, Canada, L8S 4K1 (2002)
29. Nedialkov, N.S., Jackson, K.R., Corliss, G.F.: Validated solutions of initial value problems for ordinary differential equations. Appl. Math. Comp. **105**(1), 21–68 (1999)
30. Neumaier, A.: Interval Methods for Systems of Equations. Cambridge University Press, Cambridge (1990)
31. Parnas, D.L.: Inspection of safety-critical software using program-function tables. In: Software fundamentals: collected papers by David L. Parnas, pp. 371–382. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (2001)
32. Patrikalakis, N.M., Maekawa, T., Ko, K.H., Mukundan, H.: Surface to surface intersection. In: L. Piegl (ed.) International CAD Conference and Exhibition, CAD'04. Thailand (2004)
33. Pharr, M., Humphreys, G.: Physically Based Rendering: From Theory to Implementation. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2004)
34. Ramdani, N., Meslem, N., Raïssi, T., Candau, Y.: Set-membership identification of continuous-time systems. In: B. Ninness, H. Hjalmarsson (eds.) 14th IFAC Symposium on System Identification, 2006, vol. 14. IFAC (2007)

35. Rump, S.: Verification methods: Rigorous results using floating-point arithmetic. In: Acta Numerica, pp. 287–449. Cambridge University Press (2010)
36. Schrod, J.: The cweb class. CTAN, the Comprehensive TEX Archive Network (1995)
37. Schrod, J.: Typesetting CWEAVE output. CTAN, the Comprehensive TEX Archive Network (1995)
38. Smith, L.M.C., Samadzadeh, M.H.: An annotated bibliography of literate programming. ACM SIGPLAN Notices **26**(1), 14–20 (1991)
39. Stauning, O.: Automatic validation of numerical solutions. Ph.D. thesis, Technical University of Denmark, DK-2800, Lyngby, Denmark (1997)
40. Stauning, O., Bendtsen, C.: FADBAD++ web page (2003). http://www.imm.dtu.dk/fadbad.html

Chapter 2
A New Method for Inner Estimation of Solution Sets to Interval Linear Systems

Sergey P. Shary

Abstract For an interval system of linear equations $\boldsymbol{A}x = \boldsymbol{b}$, we consider the problem of inner estimation of its solution set, formed by all the solutions to point systems $Ax = b$ with $A \in \boldsymbol{A}$ and $b \in \boldsymbol{b}$. The so-called "center approach" to the problem is developed when the inner interval box is constructed around an a priori known center point from the solution set. Determining the size of the inner box is shown to be reduced to a maximization problem for a special quasiconcave objective function.

2.1 Introduction

In our work, we consider interval linear equations systems of the form

$$\left\{\begin{array}{ccccccccc} \boldsymbol{a}_{11}x_1 & + & \boldsymbol{a}_{12}x_2 & + & \ldots & + & \boldsymbol{a}_{1n}x_n & = & \boldsymbol{b}_1, \\ \boldsymbol{a}_{21}x_1 & + & \boldsymbol{a}_{22}x_2 & + & \ldots & + & \boldsymbol{a}_{2n}x_n & = & \boldsymbol{b}_2, \\ \vdots & & \vdots & & \ddots & & \vdots & & \vdots \\ \boldsymbol{a}_{m1}x_1 & + & \boldsymbol{a}_{n2}x_2 & + & \ldots & + & \boldsymbol{a}_{mn}x_n & = & \boldsymbol{b}_m, \end{array}\right. \tag{2.1}$$

or, briefly,

$$\boldsymbol{A}x = \boldsymbol{b}, \tag{2.2}$$

where $\boldsymbol{A} = (\boldsymbol{a}_{ij})$ is an interval $m \times n$-matrix and $\boldsymbol{b} = (\boldsymbol{b}_i)$ is an interval m-vector. The above interval systems are understood as families of usual "point" linear systems $Ax = b$ with the same structure, while the matrices A are taken from $\boldsymbol{A}$ and the vectors b are taken from $\boldsymbol{b}$.

It is well-known that, for interval equations systems, solutions and solution sets can be defined in various ways (see e.g. [15–18]), but below we confine ourselves

Sergey P. Shary
Institute of Computational Technologies SB RAS, 6 Lavrentiev ave., 630090 Novosibirsk, Russia
e-mail: shary@ict.nsc.ru

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_2,

only to the so-called *united solution set* for (2.1)–(2.2), the set formed by solutions x to the point systems $Ax = b$ with the matrix A and right-hand side vector b independently varying through $\boldsymbol{A}$ and $\boldsymbol{b}$ respectively. The united solution set is rigorously defined as

$$\varXi(\boldsymbol{A},\boldsymbol{b}) = \{\, x \in \mathbb{R}^n \mid (\exists A \in \boldsymbol{A})(\exists b \in \boldsymbol{b})(Ax = b)\,\}, \qquad (2.3)$$

and it is called just *solution set* for (2.1)–(2.2) in the rest of the paper, insofar as the other solution sets are not treated herein.

The solution set $\varXi(\boldsymbol{A},\boldsymbol{b})$ is known to be a polyhedral set, generally nonconvex, while its intersection with each orthant of the space $\mathbb{R}^n$ is convex. An exact description of the solution set may grow exponentially as the dimension n increases, thus being practically impossible even for several tens of unknowns. On the other hand, in most real-life problem statements such an exact description of the solution set is not necessary. The practice is usually satisfied by an *estimate* of the solution set, i.e. an approximate description that meets the requirements of the problem under solution.

In this work, we are interested in computing *inner* interval estimates (subsets) for the solution set $\varXi(\boldsymbol{A},\boldsymbol{b})$, i. e. we solve the following problem:

$$\boxed{\begin{array}{c}\text{Find a box } \boldsymbol{U} \text{ (as wide as possible)}\\ \text{contained in the solution set } \varXi(\boldsymbol{A},\boldsymbol{b})\\ \text{of the interval linear system}\boldsymbol{A}x = \boldsymbol{b}.\end{array}} \qquad (2.4)$$

There are several known approaches to solving the problem of inner interval estimation of the solution sets to interval linear systems proposed in the literature. Among those, the so-called formal (algebraic) approach is especially efficient for square (i.e., with $m = n$) interval linear systems, developed in [8, 15, 17, 18]. Nonetheless, for arbitrary interval linear systems with rectangular (non-square) matrices, i.e. when $m \neq n$, inner interval estimation of the solution sets is an actual and significant problem. Relying on vivid geometric considerations, we propose a simple and quite general technique for constructing a box inscribed into $\varXi(\boldsymbol{A},\boldsymbol{b})$ around an a priori known point from this set (see Fig. 2.1). It is shown that the considered problem reduces to computing maximum of a special quasiconcave function, and its approximate value can be obtained by elementary means.

In the rest of the paper, we do not require regularity properties for $\boldsymbol{A}$ and even admit the case of unbounded solution set $\varXi(\boldsymbol{A},\boldsymbol{b})$. The only mild condition on $\boldsymbol{A}$ is that it must not have entirely zero rows.

Our notation follows the well-known project of informal international standard [6]. In particular, intervals and interval quantities are denoted by boldface letters — $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \ldots, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z},$ — while non-interval (point) objects are not distinguished in any way. Arithmetical operations with the interval quantities are those of the classical interval arithmetic $\mathbb{IR}$ (see, e.g., [1, 9, 10]). Underlining and overlining — $\underline{\boldsymbol{a}}$, $\overline{\boldsymbol{a}}$ — denote lower and upper endpoints of the interval $\boldsymbol{a}$, and, additionally,

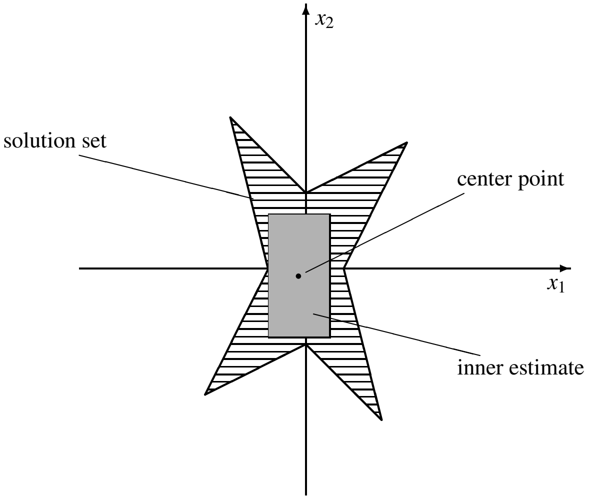


Fig. 2.1: Inner estimation of the solution set.

$\operatorname{mid} \boldsymbol{a} = \frac{1}{2}(\overline{\boldsymbol{a}} + \underline{\boldsymbol{a}})$ — midpoint (center) of the interval,

$\operatorname{rad} \boldsymbol{a} = \frac{1}{2}(\overline{\boldsymbol{a}} - \underline{\boldsymbol{a}})$ — radius of the interval,

$|\boldsymbol{a}| = \max\{\,|\overline{\boldsymbol{a}}|, |\underline{\boldsymbol{a}}|\,\}$ — absolute value (modulus) of the interval,

$\langle \boldsymbol{a} \rangle = \begin{cases} \min\{\,|\overline{\boldsymbol{a}}|, |\underline{\boldsymbol{a}}|\,\}, & \text{if } 0 \notin \boldsymbol{a}, \\ 0, & \text{otherwise}, \end{cases}$ — mignitude of the interval (antipode of the absolute value), the smallest distance between its points and zero.

With respect to interval vectors and matrices, the operations of taking the midpoint, radius and absolute value are applied in component-wise and element-wise manner.

We expect that the reader is familiar with fundamentals of interval analysis, e.g. from the books [1,9,10].

2.2 Refinement of Problem Statement

In applications, the problem statement (2.4) often contains additional information about the desired form of the box $\boldsymbol{U} = (\boldsymbol{U}_1, \boldsymbol{U}_2, \dots, \boldsymbol{U}_n)^\top$ that has to estimate $\Xi(\boldsymbol{A}, \boldsymbol{b})$ from inside: the widths of the components of $\boldsymbol{U}$ are supposed to be proportional to the respective components of a real positive vector

$$w = (w_1, w_2, \ldots, w_n), \qquad w_i > 0\,.$$

In other words, the formulation (2.4) is additionally supplied with the weight coefficients w_i for the widths (or radii) of the components of the inner box $\boldsymbol{U}$, such that

$$\mathrm{rad}\,\boldsymbol{U}_i / \mathrm{rad}\,\boldsymbol{U}_j = w_i / w_j, \qquad i, j = 1, 2, \ldots, n.$$

Scaling the interval system (2.1)–(2.2) by the nonsingular diagonal matrix

$$W = \mathrm{diag}\{w_1, w_2, \ldots, w_n\}$$

with the entries w_1, w_2, …, w_n along the main diagonal can reduce the problem to the simplest case when $w = (1, 1, \ldots, 1)$ and the box $\boldsymbol{U}$ turns to a cube that we have to inscribe into the solution set of a modified interval equations system. Moreover, we have

Proposition. *Let $\tilde{\boldsymbol{A}} = \boldsymbol{A}W$. The interval vector $\tilde{\boldsymbol{U}}$ with equal component widths, i.e. such that*

$$\mathrm{rad}\,\tilde{\boldsymbol{U}}_i = \mathrm{rad}\,\tilde{\boldsymbol{U}}_j, \qquad i, j = 1, 2, \ldots, n,$$

is a solution of the inner estimation problem (2.4) *for the modified interval system $\tilde{\boldsymbol{A}}x = \boldsymbol{b}$ if and only if the interval vector $\boldsymbol{U} = W\tilde{\boldsymbol{U}}$ with the desired ratios of the component widths is a solution to the inner estimation problem* (2.4) *for the original system $\boldsymbol{A}x = \boldsymbol{b}$.*

Proof. We use Beeck's characterization [10] of the solution set to the interval linear system (2.1)–(2.2): for $x \in \mathbb{R}^n$

$$x \in \Xi(\boldsymbol{A}, \boldsymbol{b}) \qquad \Longleftrightarrow \qquad \boldsymbol{A}x \cap \boldsymbol{b} \neq \varnothing. \tag{2.5}$$

In particular, for the modified equations system

$$\tilde{x} \in \Xi(\tilde{\boldsymbol{A}}, \boldsymbol{b}) \qquad \Longleftrightarrow \qquad \tilde{\boldsymbol{A}}\tilde{x} \cap \boldsymbol{b} \neq \varnothing. \tag{2.6}$$

Multiplication by the matrix W defines a one-to-one correspondence between the points of the boxes $\boldsymbol{U}$ and $\tilde{\boldsymbol{U}}$ according to the rule

$$x \rightleftarrows \tilde{x} = Wx$$

for $x \in \boldsymbol{U}$ and $\tilde{x} \in \tilde{\boldsymbol{U}}$. Further, for every pair of the mutually corresponding x and $\tilde{x}$, there holds

$$\boldsymbol{A}x = \boldsymbol{A}WW^{-1}x = \tilde{\boldsymbol{A}}\tilde{x},$$

so that the relations from the right-hand sides of the equivalences (2.5) and (2.6) either fulfill or not fulfill simultaneously. Moreover, for each $i, j = 1, 2, \ldots, n$, we really have

$$\mathrm{rad}\,\boldsymbol{U}_i / \mathrm{rad}\,\boldsymbol{U}_j = w_i / w_j,$$

as was required.

To sum up, in the rest of the paper we can consider the inner estimation problem (2.4) with the additional requirement that the interval vector $\boldsymbol{U}$ should have equal component widths.

2.3 Idea of our Approach

If we find a point from the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$, then it can be further used as a "center" around which the interval solution to the problem (2.4) is to be constructed somehow, by "inflation" etc. (see Fig. 2.1). This is the main idea of the approach developed, so that one can call it "center approach" in analogy to what has been done in [4, 16] for the inner estimation of the *tolerable solution set*. So,

- we look for a point $t \in \Xi(\boldsymbol{A},\boldsymbol{b})$ first,
- then we use the coordinates of t for the computation of the size of the inner estimating cube with the center in t.

The formula for the size of the interval solution of the problem (2.4) is going to be derived later (see Section 2.5). Computation according to this formula involves taking maximum of a rational expression with moduli over a box, so that the entire solution of the inner estimation problem (2.4) boils down to an optimization over a box provided that a point $t \in \Xi(\boldsymbol{A},\boldsymbol{b})$ is known. We consider this in Section 2.6 in details.

2.4 Choosing Center of Inner Estimate

The problem of recognition of whether the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ is empty or not and the problem of finding a point from the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ are known to be NP-hard in general [7]. A universal method for solving these problems can exploit the fact that intersections of the solution sets to interval linear systems with every orthant of the space $\mathbb{R}^n$ are convex polyhedral sets whose boundary planes are described by equations one can easily write out from the interval matrix and right-hand side vector of the system (see, e.g., [3, 11]). Therefore, finding out whether the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ has empty or nonempty intersection with each orthant of $\mathbb{R}^n$ can be revealed by developed linear programming techniques. Overall, the recognition of the solution sets to interval linear systems and finding a point from it requires no more than 2^n solutions of linear inequalities systems, and this result cannot be principally improved.

Therefore, in the general situation, finding a point from the solution set and its adjustment are not easy tasks. It makes sense to give a list of particular prescriptions for the solution of the above problems in some specific cases.

We consider first a square interval system with an $n \times n$-matrix $\boldsymbol{A}$. If it is regular (i.e., all $A \in \boldsymbol{A}$ are not singular), then the point t from $\Xi(\boldsymbol{A},\boldsymbol{b})$ can be obtained as the result of solving a point linear system $At = b$ with A from $\boldsymbol{A}$ and b from $\boldsymbol{b}$, say, the "middle system"

$$(\operatorname{mid}\boldsymbol{A})\,t = \operatorname{mid}\boldsymbol{b}.$$

Checking regularity of the interval matrix $\boldsymbol{A}$ can be performed by the techniques proposed e.g. in [12].

Let us consider now the case of a singular interval matrix $\boldsymbol{A}$, that is, when it contains a singular point matrix. It is well-known that the set of singular matrices forms a smooth manifold with co-dimension 1 in the set of all $n \times n$-matrices, thus being quite a meager set with zero Lebesgue measure in $\mathbb{R}^{n\times n}$. Hence, if all the entries of the matrix $\boldsymbol{A}$ have nonzero widths, then we can always hope to arrive at a regular point matrix A as the result of proper varying entries of the point $n \times n$-matrix within $\boldsymbol{A}$. Again, it suffices to solve the system $At = b$ with any $b \in \boldsymbol{b}$ in order to find the "center" point t.

What should we do in case of rectangular equation systems? Sometimes, the technique based on the so-called *recognizing functional* may help in this case, which has been elaborated by the author in [14, 16]. We would remind some facts and concepts.

Theorem 2.1. *Let $\boldsymbol{A}$ be an interval $m\times n$-matrix, $\boldsymbol{b}$ be an interval m-vector, and the expression*

$$\operatorname{Uni}(x,\boldsymbol{A},\boldsymbol{b}) = \min_{1\le i\le m}\left\{ \operatorname{rad}\boldsymbol{b}_i - \left\langle \operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n} \boldsymbol{a}_{ij}x_j \right\rangle \right\}$$

defines a functional $\operatorname{Uni} : \mathbb{R}^n \to \mathbb{R}$. *The membership of a point x in the solution set to an interval linear system $\boldsymbol{A}x = \boldsymbol{b}$ is equivalent to nonnegativity of the functional* Uni *in x,*

$$x \in \Xi(\boldsymbol{A},\boldsymbol{b}) \qquad \Longleftrightarrow \qquad \operatorname{Uni}(x,\boldsymbol{A},\boldsymbol{b}) \ge 0,$$

i. e., the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ of the interval linear system is Lebesgue set $\{\, x \in \mathbb{R}^n \mid \operatorname{Uni}(x,\boldsymbol{A},\boldsymbol{b}) \ge 0 \,\}$ of the functional Uni.

If it is clear from the context which interval system is meant, then we shall write simply $\operatorname{Uni}(x)$ instead of $\operatorname{Uni}(x,\boldsymbol{A},\boldsymbol{b})$.

Proof. A point x belongs to the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ if and only if there exists a matrix $\tilde{A} = (\tilde{a}_{ij}) \in \boldsymbol{A}$, such that

$$\tilde{A}x \in \boldsymbol{b}.$$

After writing out the matrix-vector product and representing the right-hand side intervals in the center-radius form, this membership takes the from

$$\sum_{j=1}^{n} \tilde{a}_{ij}x_j \in \operatorname{mid}\boldsymbol{b}_i + \left[-\operatorname{rad}\boldsymbol{b}_i, \operatorname{rad}\boldsymbol{b}_i\right], \qquad i = 1,2,\dots,m.$$

Adding $(-\operatorname{mid} \boldsymbol{b}_i)$ to both sides of the above inclusions, we get the equivalent relations

$$\sum_{j=1}^{n} \tilde{a}_{ij} x_j \; - \operatorname{mid} \boldsymbol{b}_i \in \bigl[-\operatorname{rad} \boldsymbol{b}_i, \operatorname{rad} \boldsymbol{b}_i \bigr], \qquad i = 1, 2, \dots, m,$$

which are, in its turn, equivalent to

$$\left| \sum_{j=1}^{n} \tilde{a}_{ij} x_j \; - \operatorname{mid} \boldsymbol{b}_i \right| \leq \operatorname{rad} \boldsymbol{b}_i \;,$$

and therefore

$$\operatorname{rad} \boldsymbol{b}_i - \left| \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} \tilde{a}_{ij} x_j \right| \geq 0 \tag{2.7}$$

for every $i = 1, 2, \dots, m$.

Hence, $x \in \Xi(\boldsymbol{A}, \boldsymbol{b})$ if and only if for each index i there exist such $\tilde{a}_{ij} \in \boldsymbol{a}_{ij}$, $j = 1, 2, \dots, n$, that the inequalities (2.17) are true. This amounts to the fulfillment of

$$\max_{\substack{\tilde{a}_{ij} \in \boldsymbol{a}_{ij}, \\ j=1,2,\dots,n}} \left\{ \operatorname{rad} \boldsymbol{b}_i - \left| \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} \tilde{a}_{ij} x_j \right| \right\} \;\geq\; 0 \tag{2.8}$$

for $i = 1, 2, \dots, m$. Bringing the maximum into the brackets and taking into account that the natural interval extension of the expression under module coincides with its range of values, we get for $i = 1, 2, \dots, m$

$$\left\{ \operatorname{rad} \boldsymbol{b}_i - \left\langle \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} \boldsymbol{a}_{ij} x_j \right\rangle \right\} \;\geq\; 0 \tag{2.9}$$

instead of (2.8). Finally, taking the minimum, we can reduce m conditions (2.9) into one, to get that the point x belongs to the set $\Xi(\boldsymbol{A}, \boldsymbol{b})$ only in the case when

$$\min_{1 \leq i \leq m} \left\{ \operatorname{rad} \boldsymbol{b}_i - \left\langle \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} \boldsymbol{a}_{ij} x_j \right\rangle \right\} \;\geq\; 0,$$

as required.

One may see that the functional Uni "recognizes", through the sign of its values, whether the point is in the solution set $\Xi(\boldsymbol{A}, \boldsymbol{b})$ or not. This is why we use the term "recognizing" with respect to it. Additionally, the following properties hold [14]:

1) The functional Uni is concave in each orthant of $\mathbb{R}^n$, and if the matrix $\boldsymbol{A}$ has entirely noninterval (point) columns, then $\operatorname{Uni}(x, \boldsymbol{A}, \boldsymbol{b})$ is concave on unions of several orthants.
2) The functional $\operatorname{Uni}(x, \boldsymbol{A}, \boldsymbol{b})$ is continuous and attains a finite maximum over the whole space $\mathbb{R}^n$.

3) If $\mathrm{Uni}(x,\boldsymbol{A},\boldsymbol{b}) > 0$, then x is a point from the topological interior $\mathrm{int}\,\Xi(\boldsymbol{A},\boldsymbol{b})$ of the solution set.
4) Under some additional conditions on $\boldsymbol{A}$, $\boldsymbol{b}$ and x, the reverse is also true: the membership $x \in \mathrm{int}\,\Xi(\boldsymbol{A},\boldsymbol{b})$ implies $\mathrm{Uni}(x,\boldsymbol{A},\boldsymbol{b}) > 0$.

The last two properties of the recognizing functional enables us to use it for deciding whether a point belongs to the interior of the solution set. This is especially important inasmuch as our technique can construct a solid inner estimate of the solution set only around the center point t that lies in the interior of the solution set $\mathrm{int}\,\Xi(\boldsymbol{A},\boldsymbol{b})$.

As a consequence of the results obtained, we arrive at the following practical prescription for the correction of the point t in our "center" approach to the solution of the problem (2.4): find a starting guess and then, using gradient ascent, try reaching better value of the recognizing functional Uni. If the value found is strictly greater than zero, then we are in the interior of the solution set.

We do not discuss the question of optimization (the best choice) of the center of the inner interval box, since it is closely related to specific needs of the customers that solve a practical problem statement.

2.5 Formula for Size of Inner Estimate

Theorem 2.2. *If a point $t \in \mathbb{R}^n$ belongs to the solution set of an interval linear system $\boldsymbol{A}x = \boldsymbol{b}$, i.e. $t \in \Xi(\boldsymbol{A},\boldsymbol{b})$, then*

$$\rho = \min_{1\leq i\leq m} \max_{A\in\boldsymbol{A}} \left\{ \frac{\mathrm{rad}\,\boldsymbol{b}_i - \left| \mathrm{mid}\,\boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij}t_j \right|}{\sum_{j=1}^{n} |a_{ij}|} \right\} \geq 0 \qquad (2.10)$$

and the interval vector $\boldsymbol{U} = (t + \rho\boldsymbol{e})$, $\boldsymbol{e} = ([-1,1],\ldots,[-1,1])^\top$, with the center t is entirely contained in the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$.

The expression under extrema in (2.10) looks very impressive, but it has a clear sense which is worth mentioning. The vector $|\,\mathrm{mid}\,\boldsymbol{b} - At\,|$ is composed of absolute values of the deviations of the product At components from the center of the right-hand side of the interval linear system considered. The signs of the differences between the radii of the right-hand side and such deviations, given by the components of $(\,\mathrm{rad}\,\boldsymbol{b} - |\,\mathrm{mid}\,\boldsymbol{b} - At\,|\,)$, show whether the image At of the point t under the linear transformation A belongs to the right-hand side vector $\boldsymbol{b}$. This all is familiar to us from the previous section, where we used the same technique to derive the recognizing functional Uni. However, when divided by the sums $\sum_j |a_{ij}|$ of the moduli of the entries in the respective rows of A, the components of the vector $(\,\mathrm{rad}\,\boldsymbol{b} - |\,\mathrm{mid}\,\boldsymbol{b} - At\,|\,)$ produce a new characteristic, namely, sensitivity of the recognizing functional with respect to variations of its first argument. More precisely, the minimum of such ratios over all the rows of A gives a "perturbation robustness" that shows how much we can shift the point t in order not to leave the solution set of the interval linear system $Ax = \boldsymbol{b}$.

Proof. Since the matrix of the interval linear system does not have zero rows, then

$$\sum_{j=1}^{n} |a_{ij}| \;>\; 0$$

for every $i = 1, 2, \ldots, m$, and $\rho \geq 0$ is equivalent to nonnegativity of the expression

$$\min_{1\leq i\leq m} \max_{A\in\boldsymbol{A}} \left\{ \operatorname{rad} \boldsymbol{b}_i - \left| \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij} t_j \right| \right\},$$

which defines the values of the recognizing functional Uni in the point $t \in \mathbb{R}^n$ due to the theorem of Section 2.4. It is indeed nonnegative for $t \in \Xi(\boldsymbol{A}, \boldsymbol{b})$.

Starting the substantiation of the second statement of the theorem, suppose first that the matrix $\boldsymbol{A}$ in the problem (2.4) has zero width, i.e. is noninterval, $\boldsymbol{A} = A = (a_{ij})$. Denoting then

$$\rho_A \;=\; \min_{1\leq i\leq m} \left\{ \frac{\operatorname{rad} \boldsymbol{b}_i - \left| \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij} t_j \right|}{\sum_{j=1}^{n} |a_{ij}|} \right\}, \tag{2.11}$$

we represent every $x \in \boldsymbol{U}$ in the form $x = t + y$, where $y \in \mathbb{R}^n$ and

$$\max_{1\leq k\leq n} |y_k| \leq \rho_A.$$

In view of the fact that

$$|y_i| \;\leq\; \rho_A \;\leq\; \frac{\operatorname{rad} \boldsymbol{b}_i - \left| \operatorname{mid} \boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij} t_j \right|}{\sum_{j=1}^{n} |a_{ij}|}, \qquad i = 1, 2, \ldots, m,$$

the following inequalities chain is valid for each $i = 1, 2, \ldots, m$:

$$\begin{aligned}|\,(Ay)_i| \;&=\; \left|\,\sum_{j=1}^{n} a_{ij}y_j\right| \;\leq\; \sum_{j=1}^{n} |a_{ij}|\,|y_j| \;\leq\; \rho_A\cdot\sum_{j=1}^{n}|a_{ij}|\\ &\leq\; \operatorname{rad}\boldsymbol{b}_i - \left|\,\operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij}t_j\,\right|\\ &=\; \operatorname{rad}\boldsymbol{b}_i - \big|\operatorname{mid}\boldsymbol{b}_i - (At)_i\big|.\end{aligned}$$

As far as $Ay = Ax - At$, we get

$$(At)_i - \operatorname{rad}\boldsymbol{b}_i + \big|\operatorname{mid}\boldsymbol{b}_i - (At)_i\big| \;\leq\; (Ax)_i \;\leq\; (At)_i + \operatorname{rad}\boldsymbol{b}_i - \big|\operatorname{mid}\boldsymbol{b}_i - (At)_i\big|$$

or, which is equivalent,

$$\begin{aligned}&\underline{\boldsymbol{b}}_i - (\operatorname{mid}\boldsymbol{b}_i - (At)_i) + |\operatorname{mid}\boldsymbol{b}_i - (At)_i|\\ &\qquad\qquad \leq (Ax)_i \leq \qquad\qquad\qquad\qquad (2.12)\\ &\overline{\boldsymbol{b}}_i - (\operatorname{mid}\boldsymbol{b}_i - (At)_i) - |\operatorname{mid}\boldsymbol{b}_i - (At)_i|.\end{aligned}$$

Taking into account that

$$-z + |z| \geq 0 \quad \text{and} \quad -z - |z| \leq 0$$

for any real z, the inequality (2.12) implies for every $i = 1, 2, \ldots, m$

$$\underline{\boldsymbol{b}}_i \leq (Ax)_i \leq \overline{\boldsymbol{b}}_i,$$

i.e. $Ax \in \boldsymbol{b}$. This means that the point x is a member of the solution set to the interval linear system $Ax = \boldsymbol{b}$. So, the formula (2.10) is proved for the systems (2.1)–(2.2) with only the right-hand side being interval, not the matrix.

We suppose now that the matrix $\boldsymbol{A}$ in the interval linear system (2.1)–(2.2) is essentially interval, i.e. has nonzero width, the corresponding solution set $\varXi(\boldsymbol{A},\boldsymbol{b})$ is nonempty and $t \in \varXi(\boldsymbol{A},\boldsymbol{b})$. We consider the totality of all the systems $Ax = \boldsymbol{b}$ with point matrices $A \in \boldsymbol{A}$ and inner estimates $\boldsymbol{U}_A$ of their solution sets $\varXi(A,\boldsymbol{b})$. By virtue of the fact that

$$\varXi(\boldsymbol{A},\boldsymbol{b}) = \bigcup_{A\in\boldsymbol{A}} \varXi(A,\boldsymbol{b}),$$

the union of all or some of the inner estimates of the sets $\varXi(A,\boldsymbol{b})$ for $A \in \boldsymbol{A}$ is an inner estimate of $\varXi(\boldsymbol{A},\boldsymbol{b})$ too.

Let $\boldsymbol{U}_A$ be a cube, with the fixed center t, included in the solution set of $Ax = \boldsymbol{b}$. Clearly, such inner estimates exist not for every solution set $\varXi(A,\boldsymbol{b})$ with $A \in \boldsymbol{A}$, but only for those that contain the point t. However, the union of the inner cubes $\boldsymbol{U}_A \subseteq \varXi(A,\boldsymbol{b})$ that still exist for the given t can be found in an especially simple way: it is a cube with the same center t, its size being equal to the maximum of sizes of the cubes to be united (see Fig. 2.2). In particular, if the sizes of the cubes are

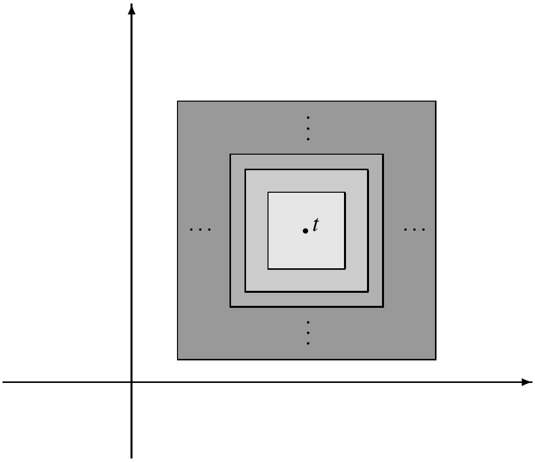


Fig. 2.2: Union of cubes with a common center is also a cube with the same center

defined by the formula (2.11), then the box

$$\boldsymbol{U} = t + \rho \boldsymbol{e}$$

is also entirely included into the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ for

$$\rho \;=\; \max_{A\in\boldsymbol{A}} \rho_A \;=\; \max_{A\in\boldsymbol{A}} \min_{1\le i\le m} \left\{ \frac{\operatorname{rad}\boldsymbol{b}_i - \left|\operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij}t_j\right|}{\sum_{j=1}^{n} |a_{ij}|} \right\}. \qquad (2.13)$$

In this expression, we have the right to take the maximum with respect to A over the whole interval matrix $\boldsymbol{A}$, no matter whether $t \in \Xi(A,\boldsymbol{b})$ or not for specific $A \in \boldsymbol{A}$. The point is that $\rho_A < 0$ in case of $t \notin \Xi(A,\boldsymbol{b})$, and such negative values of the inner minimum in the expression (2.13) in no way affect the overall nonnegative maximum of (2.13).

Finally, we can rearrange the minimum and maximum in (2.13), since, for different indices i, the expressions in the curly braces have *nonintersecting sets of arguments*, namely, they are taken over different rows of the matrix $\boldsymbol{A}$. Finally,

$$\rho \;=\; \min_{1\le i\le m} \max_{A\in\boldsymbol{A}} \left\{ \frac{\operatorname{rad}\boldsymbol{b}_i - \left|\operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij}t_j\right|}{\sum_{j=1}^{n} |a_{ij}|} \right\}.$$

This completes the proof of the theorem.

One cannot but notice a beautiful duality of the above result with the formula derived in [4, 16] for the size of inner estimate of the *tolerable solution set* to the interval linear system (2.1)–(2.2). The tolerable solution set is defined as

$$\begin{aligned}\Xi_{tol}(\boldsymbol{A},\boldsymbol{b}) &= \{x\in\mathbb{R}^n \mid (\forall A\in\boldsymbol{A})(\exists b\in\boldsymbol{b})(Ax=b)\}\\ &= \{x\in\mathbb{R}^n \mid (\forall A\in\boldsymbol{A})(Ax\in\boldsymbol{b})\}\\ &= \{x\in\mathbb{R}^n \mid \boldsymbol{A}x\subseteq\boldsymbol{b}\}\end{aligned}$$

and has many interesting practical applications (see e.g. [13, 19]). It turns out that, if $t\in\Xi_{tol}(\boldsymbol{A},\boldsymbol{b})\neq\varnothing$, then

$$\sigma \;=\; \min_{1\leq i\leq m}\;\min_{A\in\boldsymbol{A}}\left\{\frac{\operatorname{rad}\boldsymbol{b}_i - \left|\operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij}t_j\right|}{\sum_{j=1}^{n}|a_{ij}|}\right\}\;\geq\;0 \qquad (2.14)$$

and the interval vector $(t+\sigma\boldsymbol{e})$, $\boldsymbol{e}=([-1,1],\ldots,[-1,1])^\top$, is included into the tolerable solution set $\Xi_{tol}(\boldsymbol{A},\boldsymbol{b})$. Changing the logical quantifier that stands at the matrix in the definition of the solution set — from "$\exists$" to "$\forall$" — leads to changing the sense of the internal extremum in the expression (2.10) for the size of the inner box: we get minimum over $A\in\boldsymbol{A}$ instead of maximum.

An unpleasant feature of the formula (2.10) is that it produces zero, if the radius of a right-hand side component is zero. This can be partially corrected after substituting the coordinates of the center into the interval system (2.1) and transferring any interval column into the right-hand side, which acquires nonzero radius as the result.

In the expression (2.10), taking the minimum over $i\in\{1,2,\ldots,m\}$ involves no difficulties, so that the main problem in the computation of ρ is to find, for each i, the internal maximums

$$\max_{(a_{i1},\ldots,a_{in})\in(\boldsymbol{a}_{i1},\ldots,\boldsymbol{a}_{in})}\left\{\frac{\operatorname{rad}\boldsymbol{b}_i - \left|\operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n} a_{ij}t_j\right|}{\sum_{j=1}^{n}|a_{ij}|}\right\}$$

or to estimate them from below.

For further convenience, we denote the box $(\boldsymbol{a}_{i1},\boldsymbol{a}_{i2},\ldots,\boldsymbol{a}_{in})$ through

$$(\boldsymbol{X}_1,\boldsymbol{X}_2,\ldots,\boldsymbol{X}_n)=\boldsymbol{X},$$

regardless of the index $i \in \{1, 2, \ldots, m\}$, while the objective function $\mathbb{R}^n \to \mathbb{R}$, defined by the expression inside the curly braces in (2.10) and (2.14), will be denoted as

$$\Phi(x) \;=\; \frac{R - \left| M - \sum_{j=1}^{n} x_j t_j \right|}{\sum_{j=1}^{n} |x_j|}, \tag{2.15}$$

where $R = \operatorname{rad} \boldsymbol{b}_i$, $M = \operatorname{mid} \boldsymbol{b}_i$ are real constants. As the result, constructing inner interval estimate of the set $\Xi(\boldsymbol{A}, \boldsymbol{b})$ around the known center point reduces to the solution of the following optimization problem

$$\boxed{\text{Find } \max_{x \in \boldsymbol{X}} \Phi(x) \text{ or, at least, its nonnegative estimate from below.}} \tag{2.16}$$

Nonnegativity constraint is evidently implied by the practical sense of the required estimate as a radius of the inner box.

2.6 Computing Size of Inner Estimate

It is obvious that, in (2.16), the estimate of the sought-for $\max_{x \in \boldsymbol{X}} \Phi(x)$ from below may be the value the objective function $\Phi(x)$ takes at any point of the box $\boldsymbol{X}$. Therefore, if we are not going to get involved into laborious computations, then the simplest way to solve the problem (2.16) is to take maximum of the values of the objective function in several special points of its domain $\boldsymbol{X}$.

Let us denote

$$G(x) = R - \left| M - \sum_{j=1}^{n} x_j t_j \right|, \qquad H(x) = \sum_{j=1}^{n} |x_j|,$$

so that

$$\Phi(x) \;=\; \frac{G(x)}{H(x)}.$$

$G(x)$ and $H(x)$ are quite simple expressions that have only one occurrence of every variable x_j, so that their extrema over $\boldsymbol{X}$ can be easily computed as the lower and upper endpoints of the natural interval extensions $G(\boldsymbol{X})$ and $H(\boldsymbol{X})$ for the respective expressions. In particular,

$$\max_{x \in \boldsymbol{X}} G(x) \;=\; \overline{G(\boldsymbol{X})} \;=\; R - \left\langle M - \sum_{j=1}^{n} \boldsymbol{X}_j t_j \right\rangle$$

and

$$\min_{x\in\boldsymbol{X}} H(x) \;=\; \underline{H(\boldsymbol{X})} \;=\; \sum_{j=1}^{n} \langle \boldsymbol{X}_j \rangle .$$

Further, along with the values of these extrema, we can find the arguments that they deliver, tracing which of the endpoints of the intervals $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, ..., $\boldsymbol{X}_n$ produce the endpoints of the interval extensions $G(\boldsymbol{X})$ and $H(\boldsymbol{X})$ as the result of the operations with them, i.e. addition, subtraction, multiplication and taking the modulus. Overall, the simplest estimate of the solution to the problem (2.16) can be taken, for instance, as maximum of the values of the objective function $\boldsymbol{\Phi}(x)$

in the center ("most representative point") of the box $\boldsymbol{X}$,

in the point where the denominator $H(x)$ attains its minimum,

in the point where the numerator $G(x)$ attains its maximum.

If the center t of the inner box lies in the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$, then we have seen that $\max_{x\in\boldsymbol{X}} G(x) \geq 0$. So, the overall maximum of the values of $\boldsymbol{\Phi}(x)$ in the above three points is greater or equal to zero, thus satisfying the nonnegativity requirement in the formulation (16).

We turn now to more developed techniques for the solution of the optimization problem (2.16). Recall

Definition [2]. Let D be a convex set in $\mathbb{R}^n$. The function $f : D \to \mathbb{R}$ is referred to as *quasiconcave*, if for every $x, y \in D$ and $0 \leq \lambda \leq 1$ there holds

$$f\big(\lambda x + (1-\lambda) y\big) \geq \min\{\, f(x), f(y) \,\}.$$

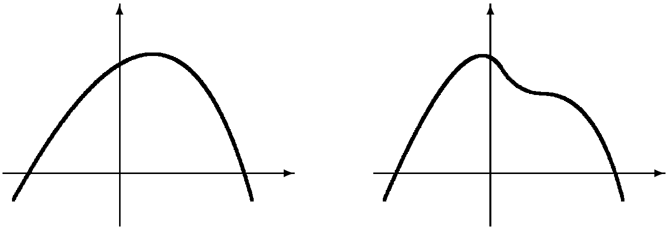


Fig. 2.3: Graphs of concave and quasiconcave functions

The function $f : D \to \mathbb{R}$ is known to be quasiconcave [2] if and only if its Lebesgue sets

$$\{\, x \in D \mid f(x) \geq \alpha \,\}$$

are convex for every $\alpha \in \mathbb{R}$ (see Fig. 2.3). In particular, a quasiconcave function cannot have several local maxima that differ in value from each other. Computing one local maximum of such functions is, at the same time, the solution of global maximization problem.

Theorem 2.3. *Let $0 \notin \boldsymbol{X} \subseteq \mathbb{R}^n$. The set $\mathfrak{D}$ of all the points from $\boldsymbol{X}$ for which the function $\Phi(x)$ defined by (2.15) takes nonnegative values is convex, and $\Phi(x)$ is quasiconcave on $\mathfrak{D}$.*

Proof. For a given level $\alpha \geq 0$, we denote through

$$S_\alpha = \{\, x \in \boldsymbol{X} \subset \mathbb{R}^n \mid \Phi(x) \geq \alpha \,\}$$

the Lebesgue set of the function $\Phi(x)$. In particular, $S_0 = \mathfrak{D}$.

If S_α is empty, there is nothing to talk about. If $S_\alpha \neq \varnothing$, then let the points x, y (not necessarily different) belong to the set S_α, so that $\Phi(x) \geq \alpha$, $\Phi(y) \geq \alpha$. Therefore,

$$R - \left| M - \sum_{j=1}^{n} x_j t_j \right| \;\geq\; \alpha \sum_{j=1}^{n} |x_j|,$$

$$R - \left| M - \sum_{j=1}^{n} y_j t_j \right| \;\geq\; \alpha \sum_{j=1}^{n} |y_j|.$$

Taking any $\lambda \in [0,1]$ and summing the above inequalities with the nonnegative weights λ and $(1-\lambda)$, we come up with the inequality of the same sense:

$$\begin{aligned} R - \lambda \left| M - \sum_{j=1}^{n} x_j t_j \right| - (1-\lambda) \left| M - \sum_{j=1}^{n} y_j t_j \right| \\ \geq \alpha \left(\lambda \sum_{j=1}^{n} |x_j| + (1-\lambda) \sum_{j=1}^{n} |y_j| \right). \end{aligned} \tag{2.17}$$

Further, applying the triangle inequality for the absolute values of intervals, we can change the left-hand side of the inequality (2.17) to a greater or equal quantity

$$R - \left| \lambda \left(M - \sum_{j=1}^{n} x_j t_j \right) + (1-\lambda) \left(M - \sum_{j=1}^{n} y_j t_j \right) \right|,$$

while the right-hand side (2.17) can be changed (due to $\alpha \geq 0$) to a smaller or equal quantity

$$\alpha \left(\sum_{j=1}^{n} |\lambda x_j + (1-\lambda) y_j| \right).$$

Finally, we have

$$R - \left| M - \sum_{j=1}^{n} (\lambda x_j + (1-\lambda) y_j) t_j \right| \;\geq\; \alpha \left(\sum_{j=1}^{n} |\lambda x_j + (1-\lambda) y_j| \right),$$

which is equivalent to

$$\Phi\bigl(\lambda x+(1-\lambda)y\bigr)\geq \alpha.$$

The point $\lambda x+(1-\lambda)y$ thus lies within the set S_α too, i.e. S_α is convex. This completes the proof of the theorem.

It is worth noting that the condition of nonnegativity on $\Phi(x)$ is not so burdensome for applications of the above result, since negativity of $\Phi(x)$ for all $x \in \boldsymbol{X}$ is only possible for uninteresting cases when the center point t does not lie within the solution set. This follows from that the negativity of $\Phi(x)$ is equivalent to negativity of the numerator in the fraction (2.15) and, hence, of the "recognizing" functional Uni in the point t (see Section 2.4). Then we have to take care of a better choice for the center point t.

The presence of moduli in the expression (2.15) makes the objective function $\Phi(x)$ nonsmooth, although it is continuous. The function is still differentiable almost everywhere over its domain of definition. Therefore, the quasiconcavity of $\Phi(x)$ may result in gradient-type methods for the solution of the problem (2.16). For instance, if $\mathrm{Pr}_{\boldsymbol{X}}$ means projection onto the box $\boldsymbol{X}$, we can apply the simplest gradient projection method

$$x^{(k+1)} := x^{(k)} + \gamma^{(k)}\,\mathrm{Pr}_{\boldsymbol{X}}\Bigl(\nabla\Phi(x^{(k)})\Bigr), \qquad k=0,1,2,\ldots, \tag{2.18}$$

with the appropriate choice of the step size $\gamma^{(k)} \in \mathbb{R}_+$ (see e.g. [2]). The components of the gradient $\nabla\Phi(x)$ are easily seen to have the form

$$\Bigl(\nabla\Phi(x)\Bigr)_i =$$

$$\frac{t_i\cdot\operatorname{sgn}\left(M-\sum_{j=1}^{n}x_jt_j\right)\cdot\left(\sum_{j=1}^{n}|x_j|\right)-\left(R-\left|M-\sum_{j=1}^{n}x_jt_j\right|\right)\cdot\operatorname{sgn}x_i}{\left(\sum_{j=1}^{n}|x_j|\right)^2},$$

$$i=1,2,\ldots,n,$$

where "sgn" means the usual sign function.

A good choice of the initial approximation $x^{(0)}$ for the process (2.18) will be a point where the objective function $\Phi(x)$ is already nonnegative. How can we find this?

As follows from the results of Section 2.4, the membership of a point t in the solution set $\Xi(\boldsymbol{A},\boldsymbol{b})$ is equivalent to

$$\mathrm{Uni}(t,\boldsymbol{A},\boldsymbol{b}) = \min_{1\leq i\leq m}\left\{\operatorname{rad}\boldsymbol{b}_i - \left\langle \operatorname{mid}\boldsymbol{b}_i - \sum_{j=1}^{n}\boldsymbol{a}_{ij}t_j\right\rangle\right\} \;\geq\; 0,$$

which, in its turn, holds true if and only the same inequality is valid for the separate i-th row of the matrix $\boldsymbol{A}$, $i = 1, 2, \ldots, m$. In terms of the function Φ defined by (2.15), this means that

$$R - \left\langle M - \sum_{j=1}^{n} \boldsymbol{X}_j t_j \right\rangle \geq 0, \tag{2.19}$$

where $\boldsymbol{X} = (\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n) = (\boldsymbol{a}_{i1}, \boldsymbol{a}_{i2}, \ldots, \boldsymbol{a}_{in})$, $R = \operatorname{rad} \boldsymbol{b}_i$, $M = \operatorname{mid} \boldsymbol{b}_i$ for a fixed index i. Therefore, to find nonnegativity points for the objective function $\Phi(x)$, we have to trace the endpoints of the intervals $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, ..., $\boldsymbol{X}_n$, at which the value of the expression

$$\left\langle M - \sum_{j=1}^{n} \boldsymbol{X}_j t_j \right\rangle$$

is attained, similar to what has been recommended in the beginning of the section. The numbers thus obtained constitute components of the sought-for starting approximation $x^{(0)}$ for the gradient ascending method (2.18).

2.7 Numerical Examples

Let us consider a numerical example with the interval linear system

$$\begin{pmatrix} [2,3] & [0,1] \\ [1,2] & [2,3] \end{pmatrix} x = \begin{pmatrix} [0,120] \\ [60,240] \end{pmatrix}, \tag{2.20}$$

proposed by E. Hansen (see [5] and earlier works). Its solution set is shown at Fig. 2.4.

In formal-algebraic approach to inner estimation of the solution set, we have to carry our considerations into Kaucher complete interval arithmetic and organize the so-called dualization equation

$$\begin{pmatrix} [3,2] & [1,0] \\ [2,1] & [3,2] \end{pmatrix} x = \begin{pmatrix} [0,120] \\ [60,240] \end{pmatrix},$$

having the matrix dualized and the right-hand side vector unchanged, and then compute its formal (algebraic) solution [8, 15, 17, 18]. It can be computed by several ways, and the most efficient subdifferential Newton method[1] in 2 iterations finds the vector

$$\begin{pmatrix} [-12,60] \\ [24,90] \end{pmatrix}. \tag{2.21}$$

[1] C-sources and executable files of its implementation for Windows are downloadable from http://www.nsc.ru/interval/shary/Codes/progr.html

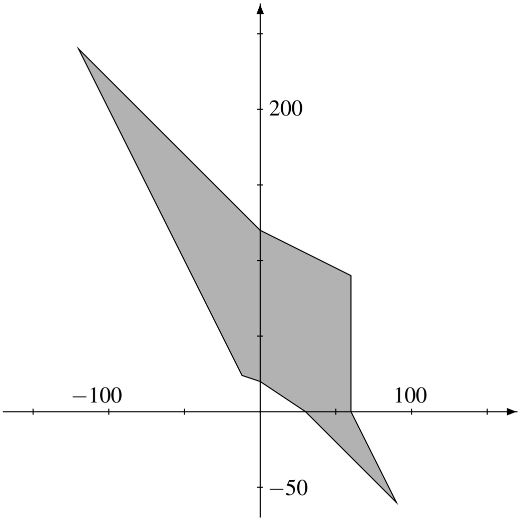


Fig. 2.4: Solution set of Hansen system (2.20)

One can make sure that this is an inclusion maximal inner estimate of the solution set for Hansen system.

Inner interval estimation with the use of our "center approach" starts from solving the midpoint system

$$\begin{pmatrix} 2.5 & 0.5 \\ 1.5 & 2.5 \end{pmatrix} x = \begin{pmatrix} 60 \\ 150 \end{pmatrix}. \tag{2.22}$$

Its solution is $(13.6364, 51.8182)^\top$ and, due to regularity of the matrix in (2.22), this vector is within the estimated solution set and can be taken as the center t of the inner box.[2]

When solving the optimization problem (2.15)–(2.16) for the first equation of the system (2.20), we have to take

$$R = 60, \qquad M = 60, \qquad \boldsymbol{X} = \big([2,3],[0,1]\big).$$

Then

$$\left\langle M - \sum_{j=1}^{2} \boldsymbol{X}_j t_j \right\rangle = 0,$$

and this value is attained at $(2.5, 0.5) \in \boldsymbol{X}$ which can serve as a starting point $x^{(0)}$ for the method (2.18).

Launched from this $x^{(0)}$, with $\Phi(x^{(0)}) = 20$, the gradient ascending (2.18) reaches the boundary of the box $\boldsymbol{X}$ at the point $\tilde{x} = (2.0, 0.631581)$ (the exact number of

[2] We keep no more than six digits in the numerical data of this section.

steps depends on the specific choice of the step size $\gamma^{(k)}$). The point $\tilde{x}$ turns out to be maximum of $\boldsymbol{\Phi}$ in $\boldsymbol{X}$ with $\boldsymbol{\Phi}(\tilde{x}) = 22.8$.

For the second equation of (2.20), the optimization problem (2.15)–(2.16) corresponds to

$$R = 90, \qquad M = 150, \qquad \boldsymbol{X} = \big([1,2],[2,3]\big).$$

We have

$$\left\langle M - \sum_{j=1}^{2} \boldsymbol{X}_j t_j \right\rangle = 0,$$

which is attained at $(1.5, 2.5)$. It is taken as the starting point $x^{(0)}$ for the method (2.18), while $\boldsymbol{\Phi}(x^{(0)}) = 22.5$. The gradient ascending (2.18) reaches the boundary of the domain box $\boldsymbol{X}$ at the point $\tilde{x} = (1.0, 2.63158)$ that delivers maximal value $\boldsymbol{\Phi}(\tilde{x}) = 24.7826$ to the objective function.

According to Theorem 2 (Section 5) and formula (2.10), we get an inner interval estimate for the solution set of Hansen system in the form

$$\begin{pmatrix} 13.6364 \\ 51.8182 \end{pmatrix} + \min\big\{22.8, 24.7826\big\} \cdot \begin{pmatrix} [-1,1] \\ [-1,1] \end{pmatrix},$$

that is,

$$\begin{pmatrix} [-9.16364, 36.4364] \\ [29.0182, 74.6182] \end{pmatrix}.$$

This is slightly worse than (2.21), but no so bad at all!

Next, we consider the interval linear system

$$\begin{pmatrix} 3.5 & [0,2] & [0,2] \\ [0,2] & 3.5 & [0,2] \\ [0,2] & [0,2] & 3.5 \end{pmatrix} x = \begin{pmatrix} [-1,1] \\ [-1,1] \\ [-1,1] \end{pmatrix}, \tag{2.23}$$

with the solution set as in Fig. 2.5 (it is shown at the jacket of the book [10], but in another projection).

Since the middle of the right-hand side vector is $(0,0,0)^\top$, the solution to the midpoint system is the zero vector too, and we can take the center of the inner box as $t = 0$. This crucially simplifies our technique, since then the numerator of the expression (2.15) does not depend on x any more. We have

$$\max_{x \in \boldsymbol{X}} \boldsymbol{\Phi}(x) = \max_{x \in \boldsymbol{X}} \left(\frac{R - |M|}{\sum_j |x_j|} \right) = \frac{R - |M|}{\min_{x \in \boldsymbol{X}} \big(\sum_j |x_j|\big)} = \frac{R - |M|}{\sum_j \langle \boldsymbol{X}_j \rangle}, \tag{2.24}$$

which is easily computable.

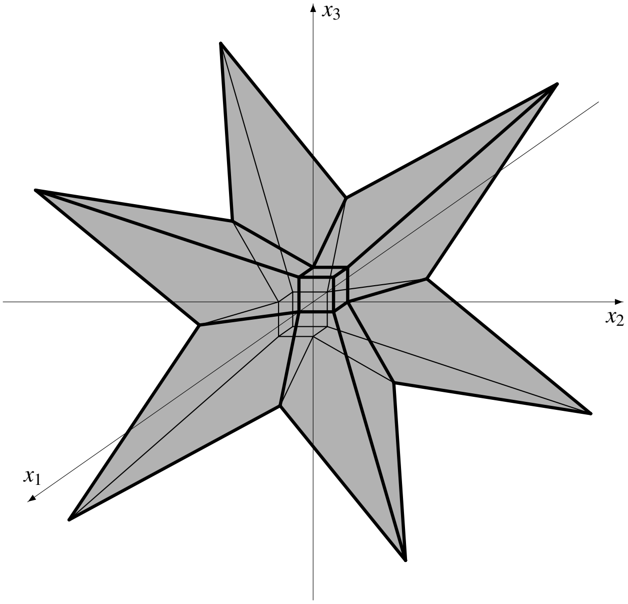


Fig. 2.5: Solution set for Neumaier system (2.23)

For the system (2.23), the expressions (2.24) taken over all three rows of the matrix coincide and equal

$$\frac{1-0}{\langle 3.5\rangle + \langle[0,2]\rangle + \langle[0,2]\rangle} = \frac{1}{3.5} = 0.285714.$$

Therefore, the inner interval box for the solution set of (2.23) should be

$$\begin{pmatrix} [-0.285714, 0.285714] \\ [-0.285714, 0.285714] \\ [-0.285714, 0.285714] \end{pmatrix}. \tag{2.25}$$

It coincides with the inner estimate obtained by formal-algebraic approach, as a proper formal solution to the interval linear system in Kaucher arithmetic

$$\begin{pmatrix} 3.5 & [2,0] & [2,0] \\ [2,0] & 3.5 & [2,0] \\ [2,0] & [2,0] & 3.5 \end{pmatrix} x = \begin{pmatrix} [-1,1] \\ [-1,1] \\ [-1,1] \end{pmatrix}.$$

The cube (2.25) is actually an inclusion maximal inner interval estimates of the solution set to (2.23) that “exhaust” its central part adjacent to the origin of coordinates.

2.8 Conclusions

The work presents a new method ("center approach") for inner interval estimation of the solution sets to interval linear systems, which is a good supplement to the earlier developed techniques.

For interval linear systems with square matrices, the quality of the results produced by the new method is slightly worse in comparison to those of formal-algebraic approach. But the new method is conceptually simpler and has wider applicability scope, being able to compute inner estimates for the solution sets to interval linear systems with general rectangular matrices. A notable feature of the "center approach" is the possibility to easily control the location of the inner box within the solution set, through changing the position of its center. Additionally, the new approach can be adapted to interval linear systems with dependencies between the entries of the matrix.

Acknowledgements The author is grateful to the referees for their valuable suggestions on improving the presentation of the paper.

References

1. Alefeld, G., Herzberger, J.: Introduction to Interval Computations. Academic Press, New York (1983)
2. Bazaraa, M.S., Shetti, C.M.: Nonlinear Programming. Theory and Algorithms. John Wiley and Sons, New York (1979).
3. Cope, J., Rust, B.: Bounds on solutions of linear systems with inaccurate data. SIAM J. Numer. Anal. **16**, 950–963 (1979)
4. Dobronets, B.S., Shaidurov, V.V.: Two-sided Numerical Methods. Nauka, Novosibirsk (1990) (in Russian)
5. Hansen, E.R., Walster, G.W.: Global Optimization Using Interval Analysis. Marcel Dekker, New York (2003)
6. Kearfott, R.B., Nakao, M.T., Neumaier, A., Rump, S.M., Shary, S.P., van Hentenryck, P.: Standardized notation in interval analysis. Comput. Technol. **15**, No. 1, 7–13 (2010) (an earlier electronic version of the paper is downloadable from URL http://www.nsc.ru/interval/INotation.pdf)
7. Kreinovich, V., Lakeyev, A., Rohn, J., Kahl, P.: Computational Complexity and Feasibility of Data Processing and Interval Computations. Kluwer, Dordrecht (1997).
8. Kupriyanova, L.: Inner estimation of the united solution set of interval linear algebraic system. Reliab. Comput. **1**, No. 1, 15–31 (1995)
9. Moore, R.E., Kearfott, R.B., Cloud, M.J.: Introduction to Interval Analysis. SIAM, Philadelphia (2009)
10. Neumaier, A.: Interval Methods for Systems of Equations. Cambridge Univ. Press, Cambridge (1990)
11. Oettli, W.: On the solution set of a linear system with inaccurate coefficients. SIAM J. Numer. Anal. **2**, No. 1, 115–118 (1965)
12. Rex, G., Rohn, J.: Sufficient conditions for regularity and singularity of interval matrices. SIAM J. Matr. Anal. Appls. **20**, 437–445 (1999)
13. Rohn, J.: Input-output planning with inexact data. Freiburger Intervall-Berichte 78/9, 1–16 (1978)

14. Shary, S.P.: On characterization of the united solution set to interval linear algebraic systems. Krasnoyarsk, 1990. 20 p. Deposited in VINITI, No. 726-B91. (in Russian)
15. Shary, S.P.: Linear static systems under interval uncertainty: algorithms to solve control and stabilization problems. In: Kreinovich, V. (ed.) Int. J. of Reliab. Comput. Supplement. Extended Abstracts of APIC'95, Int. Workshop on Applications of Interval Computations, El Paso, TX, Feb. 23-25, 1995, pp. 181–184. El Paso, University of Texas at El Paso, 1995, (an electronic version of the paper is downloadable from URL http://www.nsc.ru/interval/shary/Papers/ElPaso.pdf
16. Shary, S.P.: Solving the linear interval tolerance problem. Math. Comput. Simul. **39**, 53–85 (1995)
17. Shary, S.P.: Algebraic approach to the interval linear static identification, tolerance and control problems, or One more application of Kaucher arithmetic. Reliab. Comput. **2**, No. 1, 3–33 (1996)
18. Shary, S.P.: A new technique in systems analysis under interval uncertainty and ambiguity. Reliab. Comput. **8**, No. 5, 321–418 (2002)
19. Smagina, Ye., Brewer, I.: Using interval arithmetic for robust state feedback design. Syst. & Control Lett. **46**, 187–194 (2002)

Chapter 3
Structural Analysis for the Design of Reliable Controllers and State Estimators for Continuous-Time Dynamical Systems with Uncertainties

Andreas Rauh (✉) and Harald Aschemann

Abstract The task of designing feedforward control strategies for finite-dimensional systems in such a way that the output variables match predefined trajectories is a common goal in control engineering. Besides the widely used formulation of the corresponding system models as explicit sets of ordinary differential equations, differential-algebraic representations allow for a unified treatment of both system analysis and synthesis. For modeling and analysis of many real-life dynamic processes, differential-algebraic equations are a natural description to take into account interconnections between different physical components. Each component of such interconnected systems is described by a separate dynamic model, for instance the electric drive and the mechanical components in power trains. Moreover, side conditions are required to connect these component models by a description of power flow or, for example, geometric constraints imposed by links and joints. During system synthesis, control design tasks can be formulated in terms of initial value problems for sets of differential-algebraic equations. To check solvability, verified and non-verified algorithms are applicable which analyze the underlying system structures. The same holds for the reconstruction of internal variables and parameters on the basis of measured data. In this contribution, constructive approaches are discussed for solving both the control and estimator design using differential-algebraic formulations. It is demonstrated how these approaches can be used to show controllability and observability of dynamical systems. Numerical results for two applications conclude this paper.

Andreas Rauh
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: andreas.rauh@uni-rostock.de

Harald Aschemann
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: harald.aschemann@uni-rostock.de

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_3,

3.1 Introduction

The basis for formulating control and estimation tasks for finite-dimensional continuous-time processes is their mathematical modeling in terms of systems of ordinary differential equations (ODEs) as well as differential-algebraic equations (DAEs). Besides non-verified techniques for solving initial value problems (IVPs) for systems of DAEs (and in special cases ODEs), interval arithmetic tools are applied to analyze the structure of control and state estimation problems in a verified way.

The non-verified solver that is applied for this purpose is DAETS [20–23]. In addition to solving IVPs with consistent initial conditions on the basis of Taylor series expansions in an accurate way, DAETS provides a functionality that allows a user to structurally analyze sets of DAEs. This feature is exploited to determine approximate solutions with high quality after computing a set of consistent initial conditions for the state vector and a finite number of its time derivatives fulfilling both explicit and hidden constraints.

Moreover, verified methods are employed in this paper which rely on interval arithmetic. Basically, they were developed to quantify rounding errors in finite-precision floating-point arithmetic as well as to determine the influence of uncertainties in mathematical system models [11, 18].

In this contribution, the term *verified* solution approach is understood as a technique, implemented, for example, by using interval arithmetic software libraries, in such a way that correctness of the results is guaranteed. This means, that the computed results are represented by interval bounds containing the true solution to the equations to be solved under consideration of all possible parameter values and rounding errors. In contrast to a verified approach, *non-verified* software implementations only make use of classical finite-precision floating-point arithmetic and are, therefore, subject to inaccuracies resulting from rounding errors.

Software libraries for basic interval arithmetic functionalities such as the evaluation of arithmetic operations and functions (e.g. trigonometric and other transcendental functions) are, for instance, the C++ toolboxes PROFIL/BIAS [12] and FILIB++ [15]. In addition, most verified computational algorithms, such as those presented in this article, make use of partial derivatives of the first and higher orders as well as Taylor coefficients. Such derivatives can be obtained with the help of algorithmic differentiation [9]. The C++ library that is used for this purpose in the verified framework as well as in the solver DAETS is FADBAD++ [4].

On the basis of these software libraries, routines for verified integration of IVPs for sets of ODEs were developed. Examples for interval-based tools are VNODE-LP [19] and VALENCIA-IVP [3]. In addition, program packages such as VSPODE [16] and COSY VI [5] make use of Taylor model arithmetic to reduce the influence of overestimation. Overestimation is a general problem of verified computations. Its meaning is that enclosures of the desired solutions might get too conservative for practical purposes. It often arises if naive implementations of interval algorithms are applied.

On the one hand, packages for verified simulation of dynamical systems build the basis for offline approaches for verification, design, stability analysis and optimization of robust open-loop and closed-loop control strategies (cf. [27,29,31–33]). On the other hand, they are also applicable under certain prerequisites to the online computation of feedforward control laws as well as state and disturbance estimates.

In offline applications, interval tools are used to quantify the effects of uncertainties which result from, for example, manufacturing tolerances or measurement errors occurring unavoidably in any technical application. In the offline design, verified enclosures of *all possibly admissible* solutions of control synthesis are determined after verified enclosures of *all reachable* states have been calculated. In this case, the actual computing time is of minor importance, whereas the analysis of feasibility of all possible solutions with respect to state and control constraints is of major interest. In online applications, however, we have to fulfill given real-time requirements. For that reason, the computation is restricted to determining *only one guaranteed admissible solution* taking into account the influence of *all possible uncertainties* in such a way that constraints on state and control variables are not violated. Pessimism of the solutions introduced in some applications can be compensated effectively by implementing stabilizing feedback controllers.

In addition to directly solving IVPs for ODEs or DAEs over sufficiently short time intervals, sensitivity analysis (implemented in a verified way in VALENCIA-IVP) can be applied to solve control and, analogously, estimation problems. The sensitivity analysis provides a means for the online adjustment of control strategies. For that purpose, the sensitivity of the outputs of a dynamical system with respect to its control inputs as well as uncertain parameters can be investigated [2,30]. For further work related to the design of robust controllers, see for example [1,31,32] and the references therein.

In Section 3.2, DAE-based formulations for feedforward control synthesis as well as state and disturbance estimation are presented for finite-dimensional system models. For real-life control tasks, the structural analysis and numeric solution of these DAE systems is in the focus of this paper. Verified simulation algorithms for sets of ODEs and DAEs which are applicable to solve the corresponding IVPs are briefly summarized in Section 3.3. The strategies for feedforward control design and state estimation are applied in real-time to a finite volume representation of a distributed heating system in Section 3.4. In Section 3.5, a further tracking control task is presented for which dynamic extensions of the state equations are required. This procedure is applicable to the control of non-quasi-linear sets of DAEs and multiple-input multiple-output systems. Conclusions and an outlook on future work are given in Section 3.6.

3.2 DAE Formulation of Dynamic Systems for Feedforward Control and State Estimation

In this section, the basics of the formulation of feedforward control as well as state and disturbance estimation tasks for continuous-time dynamic systems using IVPs for sets of DAEs is given.

3.2.1 Modeling of Continuous-Time Control Objects

On the one hand, a dynamic process, that is, the plant for which control strategies are to be designed, can be described by a set of ODEs

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{p}(t), \mathbf{u}(t), t) \tag{3.1}$$

with the state vector $\mathbf{x}(t)$, a vector of constant or time-varying system parameters $\mathbf{p}(t)$, and the control vector $\mathbf{u}(t)$. In this case, all constraints for the coupling of different physical system components are eliminated explicitly such that a set of explicit ODEs $\mathbf{f}(\mathbf{x}(t), \mathbf{p}(t), \mathbf{u}(t), t)$ is obtained.

On the other hand, as already mentioned in the introduction to this paper, a natural modeling approach for complex dynamic systems is the use of separate sets of ODEs for smaller subprocesses which are connected with the help of algebraic constraints. The resulting set of DAEs is assumed to be given by

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), \mathbf{p}(t), \mathbf{u}(t), t) \\ 0 &= \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), \mathbf{p}(t), \mathbf{u}(t), t) \end{aligned} \tag{3.2}$$

in the following with the vector of algebraic state variables $\mathbf{y}(t)$.

For both types of system representations, the following two types of problems are solved in this contribution:

- Design a control strategy in such a way that output variables $\mathbf{h}(\mathbf{x}(t), \mathbf{y}(t))$ of the system match a desired dynamic behavior. Here, either the direct computation of the physical control input $\mathbf{u}(t)$ can be considered in the task of feedforward control or the design of a reference signal $\mathbf{w}(t)$ if an underlying (stabilizing) controller is designed beforehand using classical control approaches.
- Design state and parameter estimators in such a way that non-measured internal states and parameters are reconstructed on the basis of measured data.

For both types of problems, solvers for IVPs for DAEs can be employed not only to simulate the system dynamics but also to synthesize the corresponding controllers and estimators. To simplify the notation in the following, the control and estimator synthesis is described for the system model (3.1), whereas all necessary extensions to the more general DAE formulation (3.2) are highlighted. The solvability of both controller and estimator design is checked with the help of a structural analysis of the DAE formulations given in the following subsections.

3.2.2 DAE Formulation of Trajectory Planning and Tracking Control for Systems with Consistent Initial Conditions

The first task mentioned above corresponds to *trajectory planning* and computation of *feedforward control* strategies for ODE and DAE systems. In the case of trajectory planning for the system model (3.1), reference signals $\mathbf{w}(t)$ of open-loop controllers (S is open in Fig. 3.1) or closed-loop controllers (S is closed in Fig. 3.1) are calculated in such a way that the system outputs $\mathbf{y}(t)$ follow a desired time response $\mathbf{y}_d(t)$ within given tolerances. For closed-loop control, the structure and parameters of $\mathbf{u}(\hat{\mathbf{x}}, \mathbf{w})$ are assumed to be determined beforehand using classical techniques for control synthesis.

State estimation techniques can be employed in the closed loop in Fig. 3.1 to reconstruct non-measured components of $\mathbf{x}$, $\mathbf{p}$, and $\mathbf{q}$, where $\mathbf{q}$ corresponds to (uncertain) parameters of the sensor characteristics and bounded measurement noise expressed by interval parameters. The corresponding estimates $\hat{\mathbf{x}}$ can then be used as a substitute for the unknown quantities in the closed-loop control $\mathbf{u}(\hat{\mathbf{x}}, \mathbf{w})$.

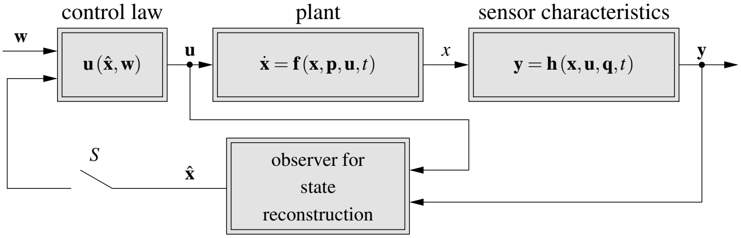

control law
plant
sensor characteristics
w
$\mathbf{u}(\hat{\mathbf{x}},\mathbf{w})$
u
$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{p},\mathbf{u},t)$
x
$\mathbf{y} = \mathbf{h}(\mathbf{x},\mathbf{u},\mathbf{q},t)$
y
S
$\hat{\mathbf{x}}$
observer for
state
reconstruction

Fig. 3.1: Observer-based closed-loop control of nonlinear dynamical systems

To determine feedforward control strategies (and reference signals), the inputs $\mathbf{u}(t)$ (and $\mathbf{w}(t)$) are computed as components of the vector $\mathbf{y}(t)$ of algebraic state variables of a set of DAEs, see also (3.2). This set of DAEs is obtained by adding the specification of the desired system outputs

$$0 = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{q}(t), t) - (\mathbf{y}_d(t) + \mathbf{y}_{tol}(t)) \qquad (3.3)$$

as an algebraic constraint $\mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), t)$ to the system model (3.1). This problem can be formulated analogously if the original system models is already a set of DAEs (3.2).

In the constraints (3.3), $[\mathbf{y}_{tol}(t)]$ represents worst-case interval bounds for the tolerances $\mathbf{y}_{tol}(t)$ between the actual and desired outputs $\mathbf{y}(t)$ and $\mathbf{y}_d(t)$. The resulting DAE system can be solved either with the help of DAETS for a point-valued approximation of the uncertain parameters or mathematically rigorously by VALENCIA-IVP for the interval parameters given above, see also Section 3.3. In both cases, the results are the control sequence $\mathbf{u}(t)$ and the state trajectories $\mathbf{x}(t)$ which are consistent with the output specification (3.3). Considering time-varying tolerances instead of fixed tolerance values is useful to express variable accuracies of control strategies for transient and steady-state operating conditions. In the case of interval parameters $\mathbf{p} \in [\mathbf{p}]$, $\mathbf{q} \in [\mathbf{q}]$, and $[\mathbf{y}_{tol}(t)] \neq [0\,;\,0]$, the application of verified solution procedures such as VALENCIA-IVP provides guaranteed enclosures for all **x** and **u** that are consistent with (3.1) and (3.3). Therefore, the results can be used directly to verify the admissibility of the solutions with respect to guaranteed compliance with state and control constraints.

Compared with approaches based on symbolic formula manipulation which can be applied to feedforward control of nonlinear exactly input-to-state linearizable sets of ODEs (as a special case of differentially flat systems) [7, 17], the interval-based approaches provided by VALENCIA-IVP are more flexible. First, uncertainties and robustness requirements can be expressed directly in the constraints (3.3) which is also not possible if a non-verified DAE solver is applied. In addition, the verified approach can also handle differentially non-flat systems if stability of the internal dynamics can be guaranteed [6,32]. For most of these non-flat systems, the output $\mathbf{y}(t)$ does not coincide exactly with $\mathbf{y}_d(t)$. However, verified techniques still allow us to compute control sequences (if they exist) for which the tolerances $[\mathbf{y}_{tol}(t)] \neq [0\,;\,0]$ in (3.3) are not violated.

3.2.3 DAE Formulation of State Estimation Tasks

Since most control structures rely on *estimates* for non-measured states, parameters, and disturbances, the DAE approach described above has to be extended. This extension leads to a one-stage procedure instead of the two-stage method used in most classical interval observers. In these two-stage approaches, the non-measured quantities are reconstructed in a filter step by solving the measurement equations for the same number of variables as linearly independent measurements (cf. [13]). In a subsequent stage, this estimate is predicted over time using a verified ODE or DAE solver up to the point at which the next measured data are available.

To estimate non-measured quantities in a one-stage DAE-based approach, the equation

$$\mathbf{q}(\mathbf{x}) = \left[\mathbf{y}_m^T \; \dot{\mathbf{y}}_m^T \; \dots \; \mathbf{y}_m^{(n_x-1)^T}\right]^T = \left[\mathbf{h}(\mathbf{x})^T \; L_{\mathbf{f}}\mathbf{h}(\mathbf{x})^T \; \dots \; L_{\mathbf{f}}^{n_x-1}\mathbf{h}(\mathbf{x})^T\right]^T \tag{3.4}$$

describing the measured variables $\mathbf{y}_m(t)$ and their i-th derivatives $\mathbf{y}_m^{(i)}(t)$ has to be solved for the state vector $\mathbf{x}(t) \in \mathbb{R}^{n_x}$, usually under the assumption of piecewise constant control $\mathbf{u}(t)$.

In (3.4), $\mathbf{y}_m^{(i)}(t)$ is expressed as the Lie derivative

$$L_{\mathbf{f}}^{i}\mathbf{h}(\mathbf{x}) = L_{\mathbf{f}}\left(L_{\mathbf{f}}^{i-1}\mathbf{h}(\mathbf{x})\right) \;, \; i = 0, \dots, n_x - 1 \;, \tag{3.5}$$

of the output $\mathbf{h}(\mathbf{x})$ along the vector field $\mathbf{f}(\mathbf{x})$ with

$$L_{\mathbf{f}}^{0}\mathbf{h}(\mathbf{x}) = \mathbf{h}(\mathbf{x}) \quad \text{and} \quad L_{\mathbf{f}}\mathbf{h}(\mathbf{x}) = \frac{\partial}{\partial \mathbf{x}}\mathbf{h}(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}) \;. \tag{3.6}$$

For a comprehensive summary of Lie algebra and its application to different control tasks, see [10].

The equation (3.4) can be solved (at least locally) for $\mathbf{x}$, if the observability matrix

$$\mathbf{Q}(\mathbf{x}) = \left[\mathbf{Q}_0^T(\mathbf{x})\ \mathbf{Q}_1^T(\mathbf{x})\ \ldots\ \mathbf{Q}_{n_x-1}^T(\mathbf{x})\right]^T \tag{3.7}$$

with $\mathbf{Q}_i(\mathbf{x}) = \frac{\partial}{\partial \mathbf{x}} L_{\mathbf{f}}^i \mathbf{h}(\mathbf{x})$, corresponding to the Jacobian of $\mathbf{q}(\mathbf{x})$ with respect to the state vector $\mathbf{x}$, has the full rank n_x. The rank of $\mathbf{Q}(\mathbf{x})$ yields sufficient information about the dimension of the observable manifold of the dynamical system [2,32].

For state, parameter, and disturbance estimation using a DAE formulation, the system's output equation $\mathbf{y}_m(t) = \mathbf{h}(\mathbf{x}(t))$ is included in the system model as a further time-dependent algebraic constraint $\mathbf{g}(\cdot)$ with interval uncertainties of the measured variables and their derivatives. In that way, the estimation task can be solved by the same procedures that are required to solve the feedforward control problem formulated in Subsection 3.2.2. Moreover, the Lie derivatives required in (3.4) coincide directly with the hidden constraints to be evaluated by the DAE solver, see also Section 3.3. These constraints are evaluated in each time step in which the DAE solver is used to integrate the dynamical system model by solving the corresponding IVP. If the interval-based solver VALENCIA-IVP is used for this purpose, the influence of measurement uncertainties on the quality of state estimates can be quantified directly by determining guaranteed consistent state enclosures.

3.2.4 Relations to Sensitivity-Based Predictive Control

As an alternative to directly solving IVPs to DAEs in control and estimator synthesis, the computation of differential sensitivities of state trajectories with respect to piecewise constant control inputs and parameters can be used [24,28]. As shown in this subsection, a sensitivity analysis provides further insight into the system structure and gives information about controllability of dynamic systems on the boundaries of the admissible operating regions.

Consider a finite-dimensional dynamical system model described by the ODEs

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \xi) \tag{3.8}$$

with the state vector $\mathbf{x} \in \mathbb{R}^n$ and the parameter vector $\xi \in \mathbb{R}^m$. In this section, the vector ξ is assumed to consist of both system parameters $\mathbf{p}$ and control inputs $\mathbf{u}$ which are constant for each time interval $t \in [t_k\,;\,t_{k+1}]$, $0 \le t_k < t_{k+1}$.

The sensitivity of the solution $\mathbf{x}(t)$ of the set of ODEs (3.8) with respect to a *time-invariant* parameter vector ξ is computed by solving an IVP for the ODEs

$$\dot{\mathbf{s}}_i(t) = \frac{\partial \mathbf{f}(\mathbf{x}(t), \xi)}{\partial \mathbf{x}} \cdot \mathbf{s}_i(t) + \frac{\partial \mathbf{f}(\mathbf{x}(t), \xi)}{\partial \xi_i} \tag{3.9}$$

with the sensitivities $\mathbf{s}_i(t) := \frac{\partial \mathbf{x}(t)}{\partial \xi_i} \in \mathbb{R}^n$ for all $i = 1, \ldots, m$.

This set of ODEs is coupled with the state equations (3.8). To determine a unique solution for the corresponding IVP for $t \geq t_k$ with the initial conditions $\mathbf{x}(t_k)$, the corresponding initial values for the sensitivities have to be determined according to

$$\mathbf{s}_i(t_k) = \frac{\partial \mathbf{x}(t_k, \xi)}{\partial \xi_i} \ . \tag{3.10}$$

Two practically important special cases are initial conditions $\mathbf{x}(t_k)$ which do not depend on the parameters ξ leading to $\mathbf{s}_i(t_k) = 0$ and sensitivity analysis with respect to the initial conditions $\mathbf{x}(t_k)$ itself leading to

$$\mathbf{s}_i(t_k) = \mathbf{e}_i \qquad \text{and} \qquad \frac{\partial \mathbf{f}(\mathbf{x}(t), \xi)}{\partial \xi_i} = 0 \ , \tag{3.11}$$

where $\mathbf{e}_i$ denotes the i-th unit vector.

The ODEs (3.9) for $\mathbf{s}_i(t)$, $t \geq t_k$, are then evaluated along the trajectories of the system states $\mathbf{x}(t)$ for $t \geq t_k$ determined by an appropriate IVP solver. All partial derivatives required in (3.9) can be computed efficiently by algorithmic differentiation provided by FADBAD++.

To use the sensitivities $\mathbf{s}_i(t)$ for the computation of dynamic feedforward control laws and for the adaptation of the parameterization of feedback controllers, the error measure

$$J = \sum_{i=k}^{k+N} \mathscr{D}(\mathbf{y}(t_i) - \mathbf{y}_d(t_i)) \tag{3.12}$$

is determined, where $\mathbf{y}(t) - \mathbf{y}_d(t)$ represents the control error.

In the following, the quadratic error measure

$$\mathscr{D}(\mathbf{y}(t_i) - \mathbf{y}_d(t_i)) := (\mathbf{y}(t_i) - \mathbf{y}_d(t_i))^T \mathbf{P} (\mathbf{y}(t_i) - \mathbf{y}_d(t_i)) \tag{3.13}$$

with the positive definite matrix $\mathbf{P} = \mathbf{P}^T$ is considered. Since the vector $\mathbf{y}(t)$ can be expressed as a function of the states $\mathbf{x}$ and the control $\mathbf{u}$ (which is assumed to be piecewise constant for $t_k \leq t < t_{k+1}$) according to

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \ , \tag{3.14}$$

the corresponding differential sensitivity of J can be determined using algorithmic differentiation with the help of (3.9) and the condition $\frac{\partial \mathbf{x}(t_{k-1})}{\partial \Delta \mathbf{u}} = 0$ by

$$\frac{\partial J}{\partial \Delta \mathbf{u}} = \sum_{i=k}^{k+N} 2 (\Delta \mathbf{g}(\mathbf{x}, \mathbf{u}))^T \cdot \mathbf{P} \cdot (\mathbf{y}(t_i) - \mathbf{y}_d(t_i)) \tag{3.15}$$

with

$$\Delta \mathbf{g}(\mathbf{x}, \mathbf{u}) := \left(\frac{\partial \mathbf{g}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \frac{\partial \mathbf{x}(t_i)}{\partial \Delta \mathbf{u}} + \frac{\partial \mathbf{g}(\mathbf{x}(t_i), \mathbf{u})}{\partial \Delta \mathbf{u}} \right) \ . \tag{3.16}$$

Here, the terms $\frac{\partial \mathbf{g}}{\partial \mathbf{x}}$ and $\frac{\partial \mathbf{g}}{\partial \Delta \mathbf{u}}$ denote the Jacobians of the output equations with respect to $\mathbf{x}$ and $\mathbf{u}$, which are evaluated for $\mathbf{x} = \mathbf{x}(t_i)$ and $\mathbf{u} = \mathbf{u}(t_{k-1})$. Here, the dimensions of the vectors $\mathbf{u}$ and $\mathbf{y}$ do not necessarily have to be identical.

Using the equations derived above, a correcting control input $\mathbf{u}(t_k)$ can be computed by

$$\mathbf{u}(t_k) = \mathbf{u}(t_{k-1}) + \Delta\mathbf{u}_k \quad \text{with} \quad \Delta\mathbf{u}_k = -\left(\frac{\partial J}{\partial \Delta \mathbf{u}}\right)^+ \cdot J \ , \tag{3.17}$$

where the superscript $+$ denotes the left pseudo inverse $\mathbf{M}^+ := \left(\mathbf{M}^T\mathbf{M}\right)^{-1}\mathbf{M}^T$ of a rectangular matrix $\mathbf{M}$. The corresponding control input $\mathbf{u}$ is used only in the time interval $t_k \leq t < t_{k+1}$ and recomputed again at $t = t_{k+1}$ with $\frac{\partial \mathbf{x}(t_k)}{\partial \Delta \mathbf{u}_{k+1}} = 0$.

This procedure can be applied to arbitrary combinations of

- the direct computation of a physical control input,
- the adaptation of gain factors of a feedback controller with a given structure, and
- the adaptation of reference signals in closed-loop control structures to improve tracking properties.

In the equations above, it is assumed that the complete state vector $\mathbf{x}$ is accessible to the controller at each time step t_k either by direct measurement or by reconstruction using a suitable observer.

An extension to sensitivity-based control of systems described by DAEs

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), \xi) \\ 0 &= \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), \xi) \end{aligned} \tag{3.18}$$

is straightforward after evaluating the corresponding sensitivity equations

$$\begin{aligned} \dot{\mathbf{s}}_{\mathbf{x},i}(t) &= \frac{\partial \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), \xi)}{\partial \mathbf{x}} \cdot \mathbf{s}_{\mathbf{x},i}(t) + \frac{\partial \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), \xi)}{\partial \mathbf{y}} \cdot \mathbf{s}_{\mathbf{y},i}(t) + \frac{\partial \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), \xi)}{\partial \xi_i} \\ 0 &= \frac{\partial \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), \xi)}{\partial \mathbf{x}} \cdot \mathbf{s}_{\mathbf{x},i}(t) + \frac{\partial \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), \xi)}{\partial \mathbf{y}} \cdot \mathbf{s}_{\mathbf{y},i}(t) + \frac{\partial \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), \xi)}{\partial \xi_i} \end{aligned} \tag{3.19}$$

with

$$\mathbf{s}_{\mathbf{x},i}(t) := \frac{\partial \mathbf{x}(t)}{\partial \xi_i} \quad \text{and} \quad \mathbf{s}_{\mathbf{y},i}(t) := \frac{\partial \mathbf{y}(t)}{\partial \xi_i} \ . \tag{3.20}$$

As shown in [25], it is possible to extend this sensitivity-based control approach using the verified ODE and DAE solvers presented in the following section in such a way that constraints on state and output variables are guaranteed not to be violated in the presence of interval uncertainties.

To verify controllability of dynamic systems on the boundary of the admissible operating range, the following criteria are checked: Reaching the maximal admissible value $\overline{y}_i$ of the system output y_i, the sensitivity $s_{y,i,j} = \frac{\partial y_i}{\partial u_j}$ has to fulfill the condition $\overline{s}_{y,i,j} < 0$ for at least one admissible control u_j. Here, the sensitivity $s_{y,i,j}$

is computed for all possible values $\mathbf{p} \in [\mathbf{p}]$ of the uncertain system parameters. If the condition $\overline{s}_{y,i,j} < 0$ is fulfilled, it is guaranteed that the critical state variable y_i remains within its admissible range. In contrast, operability is certainly not given if the violation of the state constraint cannot be avoided for any possible u_j and at least one $\mathbf{p} \in [\mathbf{p}]$ if $\underline{s}_{y,i,j} > 0$ holds. Similarly, controllability can be verified if the bound $\underline{y}_i$ is reached, where $\underline{s}_{y,i,j} > 0$ has to be guaranteed.

3.3 Verified Simulation of IVPs in ValEncIA-IVP

As shown in the previous section, IVPs for ODEs and DAEs arise naturally in the analysis and synthesis of control strategies as well as state and parameter estimation tasks. To solve IVPs for dynamic system models with uncertainties in the initial states and parameters, verified ODE and DAE solvers can be applied. In this section, the basic functionalities provided for that purpose by ValEncIA-IVP are summarized.

3.3.1 Initial Value Problems for Systems of ODEs

First, the verified solution to IVPs to the set of ODEs

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), t) \ , \ \ \mathbf{x} \in \mathbb{R}^{n_x} \tag{3.21}$$

is considered with the uncertain initial conditions $\mathbf{x}(0) \in [\mathbf{x}(0)] := [\underline{\mathbf{x}}(0)\ ;\ \overline{\mathbf{x}}(0)]$, $\underline{x}_i(0) \leq \overline{x}_i(0)$ for all $i = 1, \ldots, n_x$ with the help of the verified solver ValEncIA-IVP.

In the basic version of ValEncIA-IVP, time-varying state enclosures

$$[\mathbf{x}_{encl}(t)] := \mathbf{x}_{app}(t) + [\mathbf{R}(t)] \tag{3.22}$$

are computed iteratively which consist of a non-verified approximate solution $\mathbf{x}_{app}(t)$ with guaranteed error bounds $[\mathbf{R}(t)]$. For the sake of simplicity, we specify the iteration formulas for the ODE (3.21) in the time interval $0 \leq t \leq T$. In this case, an interval containing the derivatives $\left[\dot{\mathbf{R}}(t)\right]$ of the desired error bounds $[\mathbf{R}(t)]$ can be computed by

$$\begin{aligned} \left[\dot{\mathbf{R}}^{(\kappa+1)}(t)\right] &= -\dot{\mathbf{x}}_{app}(t) + \mathbf{f}\left(\left[\mathbf{x}^{(\kappa)}_{encl}(t)\right], t\right) \\ &= -\dot{\mathbf{x}}_{app}(t) + \mathbf{f}\left(\mathbf{x}_{app}(t) + \left[\mathbf{R}^{(\kappa)}(t)\right], t\right) =: \mathbf{r}\left(\left[\mathbf{R}^{(\kappa)}(t)\right], t\right) \end{aligned} \tag{3.23}$$

if

$$\left[\dot{\mathbf{R}}^{(\kappa+1)}(t)\right] \subseteq \left[\dot{\mathbf{R}}^{(\kappa)}(t)\right] \tag{3.24}$$

holds with

$$\left[\mathbf{R}^{(\kappa+1)}(t)\right] \subseteq \left[\mathbf{R}^{(\kappa+1)}(0)\right] + t \cdot \mathbf{r}\left(\left[\mathbf{R}^{(\kappa)}([0\,;\,t])\right], [0\,;\,t]\right) \tag{3.25}$$

and $t = T$ as well as $[\mathbf{x}(0)] \subseteq \mathbf{x}_{app}(0) + \left[\mathbf{R}^{(\kappa+1)}(t)\right]$.

In addition, we can apply the approach of computing exponential state enclosures to prevent the growth of interval diameters for asymptotically stable systems. The basic idea is to use the representation

$$[\mathbf{x}_{encl}(t)] := \exp([\Lambda] \cdot t) \cdot [\mathbf{x}_{encl}(0)] \tag{3.26}$$

for the guaranteed state enclosures with the diagonal matrix

$$[\Lambda] := \operatorname{diag}\{[\lambda_i]\} \quad , \tag{3.27}$$

where the coefficients $[\lambda_i]$ are computed iteratively by

$$\left[\lambda_i^{(\kappa+1)}\right] := \frac{f_i\left(\exp\left(\left[\Lambda^{(\kappa)}\right] \cdot [0\,;\,T]\right) \cdot [\mathbf{x}_{encl}(0)], [0\,;\,T]\right)}{\exp\left(\left[\lambda_i^{(\kappa)}\right] \cdot [0\,;\,T]\right) \cdot \left[\mathbf{x}_{encl,i}(0)\right]} \tag{3.28}$$

for all $i = 1, \ldots, n_x$ in the case of convergence, that means, for $\left[\lambda_i^{(\kappa+1)}\right] \subseteq \left[\lambda_i^{(\kappa)}\right]$.

The iteration formula (3.28) is only admissible if the value zero does not belong to the set of all reachable states in the time interval $[0\,;\,T]$. To check this property, we compute guaranteed enclosures for all states by the basic iteration formulas (3.23)–(3.25) before evaluating the tighter exponential state enclosures.

A detailed derivation of the iteration formulas of VALENCIA-IVP can be found, for example, in [3, 27]. To further tighten the computed state enclosures, consistency tests are available which exclude domains resulting from overestimation by constraints representing conservation properties such as energy balances for mechanical systems [8, 26].

3.3.2 Initial Value Problems for Systems of DAEs

As an extension to the systems considered in Subsection 3.3.1, we consider semi-explicit DAEs

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), t) \tag{3.29}$$

$$0 = \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), t) \tag{3.30}$$

with $\mathbf{f} : D \mapsto \mathbb{R}^{n_x}$, $\mathbf{g} : D \mapsto \mathbb{R}^{n_y}, D \subset \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R}^1$, and the consistent initial conditions $\mathbf{x}(0)$ and $\mathbf{y}(0)$. As for systems of ODEs, these DAEs may further depend on uncertain parameters $\mathbf{p}$. To simplify the notation, the dependency on $\mathbf{p}$ is not explicitly denoted. However, all presented results are also applicable to uncertain systems with $p_i \in \left[\underline{p}_i\,;\,\overline{p}_i\right]$, $\underline{p}_i < \overline{p}_i$, $i = 1,\ldots,n_p$. The basis for the applications in Sections 3.4 and 3.5 is the computation of guaranteed enclosures for both consistent initial conditions and solutions to IVPs for DAEs. The enclosures for the differential and algebraic variables $x_i(t)$ and $y_j(t)$, respectively, are defined by

$$[x_i(t)] := x_{app,i}(t_k) + (t - t_k)\cdot \dot{x}_{app,i}(t_k) + [R_{x,i}(t_k)] + (t - t_k)\cdot [\dot{R}_{x,i}(t)] \qquad (3.31)$$

and

$$[y_j(t)] := y_{app,j}(t_k) + (t - t_k)\cdot \dot{y}_{app,j}(t_k) + [R_{y,j}(t)] \qquad (3.32)$$

with $i = 1,\ldots,n_x$, $j = 1,\ldots,n_y$, and $t \in [t_k\,;\,t_{k+1}]$, $t_0 \le t \le t_f$.

In (3.31) and (3.32), t_k and t_{k+1} are two subsequent points of time between which guaranteed state enclosures are determined. For $t = 0$, the conditions

$$[\mathbf{x}(0)] = \mathbf{x}_{app}(0) + [\mathbf{R}_\mathbf{x}(0)] \quad \text{and} \quad [\mathbf{y}(0)] = \mathbf{y}_{app}(0) + [\mathbf{R}_\mathbf{y}(0)] \qquad (3.33)$$

have to be fulfilled with approximate solutions $\mathbf{x}_{app}(t)$ and $\mathbf{y}_{app}(t)$. They are computed, for example, by the non-verified DAE solver DAETS [20–23].

The following three-stage algorithm allows us to determine guaranteed state enclosures of a system of DAEs using the Krawczyk iteration [14] which solves nonlinear algebraic equations in a verified way.

Step 1. Compute hidden constraints that have to be fulfilled for the verified enclosures of the initial conditions $\mathbf{x}(0)$ and $\mathbf{y}(0)$ as well as for the time responses $\mathbf{x}(t)$ and $\mathbf{y}(t)$ by considering algebraic equations $g_i(\mathbf{x})$ which do not depend explicitly on $\mathbf{y}$. Differentiation with respect to time leads to

$$\frac{d^j g_i(\mathbf{x})}{dt^j} = \left(\frac{\partial L_\mathbf{f}^{j-1} g_i(\mathbf{x})}{\partial \mathbf{x}}\right)^T \cdot \mathbf{f}(\mathbf{x},\mathbf{y}) = L_\mathbf{f}^j g_i(\mathbf{x}) = 0 \qquad (3.34)$$

with $L_\mathbf{f}^0 g_i(\mathbf{x}) = g_i(\mathbf{x})$. The Lie derivatives $L_\mathbf{f}^j g_i(\mathbf{x})$ are computed automatically by using FADBAD++ [4] up to the smallest order $j > 0$ for which $L_\mathbf{f}^j g_i(\mathbf{x})$ depends on at least one component of $\mathbf{y}$.

Step 2. Compute initial conditions for the equations (3.29) and (3.30) such that the constraints (3.30) and (3.34) are fulfilled using the Krawczyk iteration.

Step 3. Substitute the state enclosures (3.31) and (3.32) for the vectors $\mathbf{x}(t)$ and $\mathbf{y}(t)$ in (3.29) and (3.30) and solve the resulting equations for $[\dot{\mathbf{R}}_\mathbf{x}(t)]$ and $[\mathbf{R}_\mathbf{y}(t)]$ with the help of the Krawczyk iteration. The hidden constraints (3.34) are employed to restrict the set of feasible solutions.

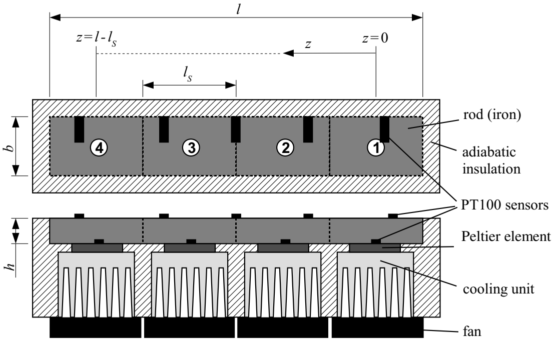
A scenario, in which this procedure is combined with a dynamic extension of the control inputs of a dynamic system using an integrator chain to obtain a uniquely solvable set of equations, is described in Section 3.5.

For the application scenarios discussed in Sections 3.4 and 3.5, the conservativeness of verified ODE and DAE solvers is not critical, since state enclosures are only computed over short time horizons. In online applications, these enclosures are further tightened by verified state estimates relying on measured data which are used to reduce overestimation.

3.4 Control of a Distributed Heating System

3.4.1 Basic Experimental Setup

To visualize the practical applicability of verified DAE solvers for feedforward control as well as state and disturbance estimation, we consider the distributed heating system in Fig. 3.2. The controlled variable of this system is the temperature at a given position of the rod. Control and disturbance inputs are provided by four Peltier elements and cooling units. The temperature $\vartheta\left(z,t\right)$ of the rod depends both on the spatial variable z and on the time t.



$\rho = 7800$ kg/m^3	density of iron
$c_p = 420$ J/(kg·K)	specific heat capacity
$\lambda = 110$ W/(m·K)	heat conductance
$\alpha = 50$ W/(m^2·K)	heat transfer coefficient
$l = 320$ mm	total length of the rod
$l_s = l/n$	length of segment
$n = 4$	number of segments
$b = 40$ mm	width of the rod
$h = 12$ mm	height of the rod
$m_S = l_s \cdot b \cdot h \cdot \rho$	mass of segment
ϑ_U	ambient temperature
$\vartheta(z,t)$	temperature profile
x_i	temperature in segment i

Fig. 3.2: Experimental setup of a distributed heating system

Mathematically, the temperature distribution is given by the parabolic PDE

$$\frac{\partial\vartheta(z,t)}{\partial t} - \frac{\lambda}{\rho c_p}\frac{\partial^2\vartheta(z,t)}{\partial z^2} + \frac{\alpha}{h\rho c_p}\vartheta(z,t) = \frac{\alpha}{h\rho c_p}\vartheta_U \tag{3.35}$$

which is discretized in its spatial coordinate into finite volume elements. Balancing of heat exchange between four volume elements leads to the ODEs

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\\ \dot{x}_4(t)\end{bmatrix} = \begin{bmatrix}a_{11} & a_{12} & 0 & 0\\ a_{12} & a_{22} & a_{12} & 0\\ 0 & a_{12} & a_{22} & a_{12}\\ 0 & 0 & a_{12} & a_{11}\end{bmatrix} \cdot \begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\\ x_4(t)\end{bmatrix} + \frac{1}{m_s c_p}\begin{bmatrix}1\\0\\0\\0\end{bmatrix} u(t) + \frac{\alpha A}{m_s c_p}\begin{bmatrix}e_1(t)\\ e_2(t)\\ e_3(t)\\ e_4(t)\end{bmatrix} \tag{3.36}$$

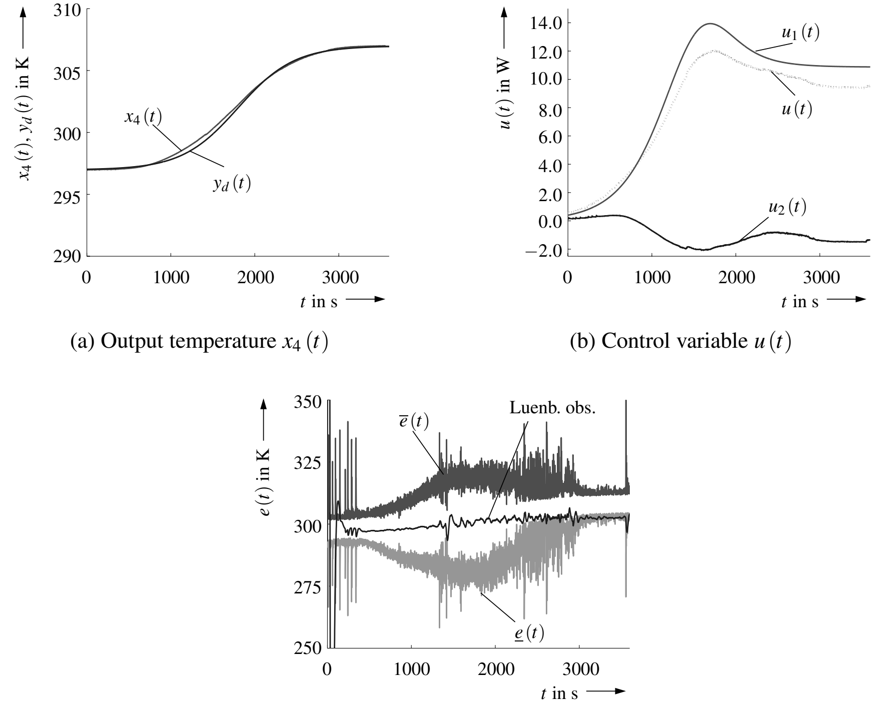
for the temperatures $x_i(t)$ in the segments $i = 1,\ldots,n = 4$ with the coefficients

$$a_{11} = -\frac{\alpha A l_s + \lambda_s b h}{l_s m_s c_p} \ , \ a_{12} = \frac{\lambda_s b h}{l_s m_s c_p} \ , \ \text{and} \ a_{22} = -\frac{\alpha A l_s + 2\lambda_s b h}{l_s m_s c_p} \ . \tag{3.37}$$

In (3.36), the input signal $u(t)$ corresponds to the heat flow into the first segment of the rod. The goal of feedforward control (determined by VALENCIA-IVP or DAETS) is the computation of an input $u(t) = u_1(t)$ in such a way that the output temperature $y(t)$ in an arbitrary segment tracks the desired temperature profile

$$y_d(t) = \vartheta_0 + \frac{(\vartheta_f - \vartheta_0)}{2} \left(1 + \tanh\left(k\left(t - \frac{3600\,\mathrm{s}}{2}\right)\right)\right) \tag{3.38}$$

with $\vartheta_0 = \vartheta_U(0)$, $\vartheta_f = \vartheta_0 + \varDelta\vartheta$, $\varDelta\vartheta = 10\,\mathrm{K}$, and $k = 0.0015$ exactly. The prediction time horizon for the DAE solver is $t_{k+1} - t_k = 1\,\mathrm{s}$. To determine a unique control, the definition $u_1(t) = 0.5 \cdot (\underline{u}_1(t) + \overline{u}_1(t))$ with $t \in [t_k; t_{k+1})$ is used.



(a) Output temperature $x_4(t)$

(b) Control variable $u(t)$

(c) Disturbance estimate $e(t)$

Fig. 3.3: Experimental results for closed-loop control of the heating system

The additive terms $e_i(t)$, $i = 1,\dots,n = 4$ summarize errors resulting from the discretization of the PDE and unmodeled disturbances which are estimated by a Luenberger observer and the DAE-based approach described in Subsection 3.2.3 implemented using VALENCIA-IVP, see the experimental results in Fig. 3.3. The interval observer detects the point of time from which on the Luenberger observer yields consistent estimates. Both estimators make use of the measured temperatures $y_{m,1} = x_1$ and $y_{m,2} = x_4$. If model errors are neglected, all e_i are equal to the ambient temperature $\vartheta_U(0)$.

For the implementation of the disturbance observer, the ODEs (3.36) are extended by $\dot{e} = 0$ with $e = e_1 = \dots = e_4$. To quantify the influence of measurement errors, the uncertainties $x_i \in y_{m,j} + [-1\ ;\ 1]\,\mathrm{K}$, $\dot{x}_i \in [-0.5\ ;\ 1.5]\,\dot{y}_{m,j}$, $i \in \{1,4\}$, $j \in \{1,2\}$ are considered in the DAE-based estimator. To compensate model errors and disturbances, output feedback $u_2(t)$ is introduced in addition to $u_1(t)$ by a PI controller

$$u_2(t) = K_I \cdot \left((y_d(t) - y(t)) + \frac{1}{T_I} \int\limits_0^t (y_d(\tau) - y(\tau))\, d\tau \right) \tag{3.39}$$

with $K_I = 3$ and $T_I = 786\,\mathrm{s}$ compensating the largest time constant T_I of the plant (3.36). Therefore, the total control input is given by $u(t) = u_1(t) + u_2(t)$.

3.4.2 Structural Analysis for Specification of Flat Outputs

For specification of the flat output

$$g(\mathbf{x},t) = x_4(t) - y_d(t) = 0 \tag{3.40}$$

of the system, the same state equations as in Subsection 3.4.1 are considered with the assumption that the error terms $e_i(t)$ are piecewise constant. In this case, the structural analysis performed in VALENCIA-IVP provides the result summarized in the following table, where only explicit dependencies on state, control, and time variables are listed.

The Lie derivative $L_{\mathbf{f}}^4 g$ corresponds to the smallest order of the derivative of the output equation $g(\mathbf{x},t)$ which is influenced directly by the control input u. Since the number of unknowns (all unknowns are marked by • in the previous scheme) and the number of hidden constraints are identical in this case, the equations $L_{\mathbf{f}}^1 g = 0, \dots, L_{\mathbf{f}}^4 g = 0$ can be solved directly by application of interval Newton techniques for the consistent states x_1, x_2, and x_3, as well as the desired control input u. Since all internal states x_i, $i = 1,\dots,4$, and the control u are uniquely defined by y_d and a finite number of its derivatives, the output $y = x_4$ corresponds to the system's flat output. Note that the value of x_4 is known a-priori by evaluation of $g = L_{\mathbf{f}}^0 g = 0$ for each point of time t, which is denoted by $\diamond$. Since the solution is uniquely defined by specification of the desired system output, no additional initial conditions are required for the synthesis of the corresponding feedforward control. However, this also means that deviations of the initial temperature distribution in the rod from the values specified by $L_{\mathbf{f}}^1 g = 0, \ldots, L_{\mathbf{f}}^4 g = 0$ inevitably lead to tracking errors $y(t) - y_d(t) \neq 0$. These deviations can be compensated by output feedback controllers according to Subsection 3.4.1.

	x_1	x_2	x_3	x_4	t	u
$\dot{x}_1$	•	•				•
$\dot{x}_2$	•	•	•			
$\dot{x}_3$		•	•	◇		
$\dot{x}_4$			•	◇		
$g(\mathbf{x},t)$				◇	◇	

	x_1	x_2	x_3	x_4	t	u
$L_{\mathbf{f}}^0 g$				◇	◇	
$L_{\mathbf{f}}^1 g$			•	◇	◇	
$L_{\mathbf{f}}^2 g$		•	•	◇	◇	
$L_{\mathbf{f}}^3 g$	•	•	•	◇	◇	
$L_{\mathbf{f}}^4 g$	•	•	•	◇	◇	•

Legend:

◇	a-priori known
•	determined via algebraic constraints of the DAE system

As an alternative to the interval-based computation of feedforward control using VALENCIA-IVP (and its structural analysis which allows us to determine guaranteed enclosures of all admissible initial conditions in a given domain for bounded control inputs u), the non-verified solver DAETS can be used if no interval uncertainties are considered for parameters and modeling errors. In Fig. 3.4, the control inputs $u(t)$ are displayed for the output defined in (3.40). For the visualization without interval uncertainties, DAETS has been used to determine the feedforward control for different variations $\Delta\vartheta = \vartheta_f - \vartheta_0$ of the output temperature.

3.4.3 Structural Analysis for Specification of Non-Flat Outputs

For specification of a non-flat output, for example

$$g(\mathbf{x},t) = x_3(t) - y_d(t) = 0 \quad , \tag{3.41}$$

the order δ of the derivative of the output equation g which is influenced directly by the control input u is smaller than the number of unknown variables. For that reason, the relative degree δ of the system is smaller than the dimension of the state vector.

Since the number of unknowns is now larger than the number of hidden constraints, the equations $L_{\mathbf{f}}^1 g = 0, \ldots, L_{\mathbf{f}}^\delta g = 0$, $\delta = 3$, cannot be solved directly by application of interval Newton techniques to obtain the missing consistent states (denoted by •) and the desired control input u. This is also demonstrated by the following result of the structural analysis.

	x_1	x_2	x_3	x_4	t	u
$\dot{x}_1$	●	●				●
$\dot{x}_2$	●	●	◇			
$\dot{x}_3$		●	◇	●		
$\dot{x}_4$			◇	●		
$g(\mathbf{x},t)$			◇		◇	

	x_1	x_2	x_3	x_4	t	u
$L_{\mathbf{f}}^0 g$			◇		◇	
$L_{\mathbf{f}}^1 g$		●	◇	●	◇	
$L_{\mathbf{f}}^2 g$	●	●	◇	●	◇	
$L_{\mathbf{f}}^3 g$	●	●	◇	●	◇	●

Therefore, to solve this system, further information about the initial conditions has to be taken into account in the following two-stage procedure. In the first stage, we identify a set of ODEs or DAEs which includes the system's output and can be solved as an IVP by specification of a suitable number of initial conditions. The resulting equations describe either an IVP for ODEs or an IVP for a set of DAEs. In the first case, all initial conditions can be specified arbitrarily. In the second case, the initial conditions have to be computed consistently with the help of the output equation $g = L_{\mathbf{f}}^0 g = 0$ and, if necessary, the lower-order constraints $L_{\mathbf{f}}^1 g = 0, \ldots, L_{\mathbf{f}}^{\tau} g = 0$, $\tau < \delta$. In the second stage, this solution to the IVP is substituted for the corresponding state variables (denoted by $\circ$) in $L_{\mathbf{f}}^{\tau+1} g = 0, \ldots, L_{\mathbf{f}}^{\delta} g = 0$. These equations, which are purely algebraic, are now solved for the remaining states (denoted by $\bullet$) and the control input $u(t)$ using interval Newton techniques.

In the following, this procedure is demonstrated for the system model (3.36) and the output specification (3.41). For specification of x_3 as the desired output (denoted by $\diamond$), it is at least necessary to know the initial temperature $x_4(0)$. Then, an IVP for the ODE for $x_4(t)$ is solved in the first stage with the known temperature profile $x_3(t)$. This information is substituted for $x_4(t)$ in the constraints $L_{\mathbf{f}}^1 g = 0, \ldots, L_{\mathbf{f}}^{\delta} g = 0$, which can now be solved for the remaining unknowns.

	x_1	x_2	x_3	x_4	t	u
$\dot{x}_1$	●	●				●
$\dot{x}_2$	●	●	◇			
$\dot{x}_3$		●	◇	○		
$\dot{x}_4$			◇	○		
$g(\mathbf{x},t)$			◇		◇	

	x_1	x_2	x_3	x_4	t	u
$L_{\mathbf{f}}^0 g$			◇		◇	
$L_{\mathbf{f}}^1 g$		●	◇	○	◇	
$L_{\mathbf{f}}^2 g$	●	●	◇	○	◇	
$L_{\mathbf{f}}^3 g$	●	●	◇	○	◇	●

Alternatively, the solution of IVPs using a DAE solver with the given initial conditions $x_2(0)$, $x_4(0)$, and the constraint $L_{\mathbf{f}}^1 g = 0$ (or the initial conditions $x_1(0)$, $x_2(0)$, $x_4(0)$, and the constraints $L_{\mathbf{f}}^1 g = 0$, $L_{\mathbf{f}}^2 g = 0$, respectively) produces the same result. The variables which are determined by the verified DAE solver in this first stage are denoted by $*$ in the following schemes. The remaining constraints $L_{\mathbf{f}}^2 g = 0$, $L_{\mathbf{f}}^3 g = 0$ (or only $L_{\mathbf{f}}^3 g = 0$, respectively), are used to compute the consistent internal system states and the input u in the stage 2 of the solution approach, denoted again by $\bullet$.

In analogy to the specification of the flat system output, the non-verified solver DAETS is applied as an alternative solution procedure. In Fig. 3.5, the corresponding feedforward control sequences are displayed for the output function (3.41).

	x_1	x_2	x_3	x_4	t	u
$\dot{x}_1$	●	$*$				●
$\dot{x}_2$	●	$*$	◇			
$\dot{x}_3$		$*$	◇	○		
$\dot{x}_4$			◇	○		
$g(\mathbf{x},t)$			◇		◇	

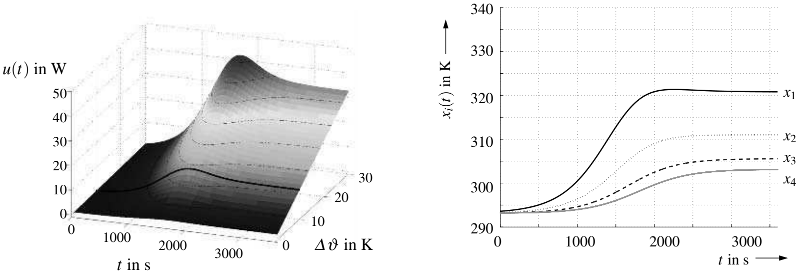
	x_1	x_2	x_3	x_4	t	u
$L_{\mathbf{f}}^0 g$			◇		◇	
$L_{\mathbf{f}}^1 g$		$*$	◇	○	◇	
$L_{\mathbf{f}}^2 g$	●	$*$	◇	○	◇	
$L_{\mathbf{f}}^3 g$	●	$*$	◇	○	◇	●

	x_1	x_2	x_3	x_4	t	u
$\dot{x}_1$	$*$	○				●
$\dot{x}_2$	$*$	○	◇			
$\dot{x}_3$		○	◇	○		
$\dot{x}_4$			◇	○		
$g(\mathbf{x},t)$			◇		◇	

	x_1	x_2	x_3	x_4	t	u
$L_{\mathbf{f}}^0 g$			◇		◇	
$L_{\mathbf{f}}^1 g$		○	◇	○	◇	
$L_{\mathbf{f}}^2 g$	$*$	○	◇	○	◇	
$L_{\mathbf{f}}^3 g$	$*$	○	◇	○	◇	●

Legend:

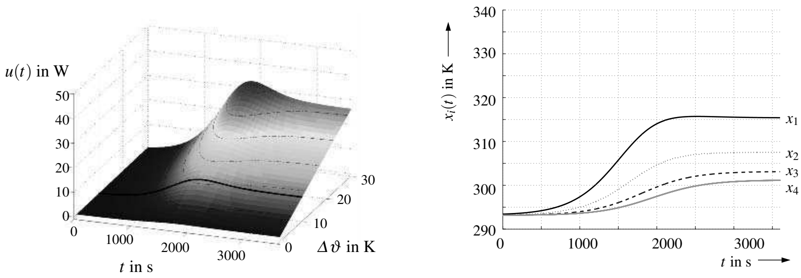
◇	a-priori known
○	determined via IVP solver (ODE/ DAE)
$*$	determined via algebraic constraints of DAE (stage 1) (not required if the flat output is specified directly)
●	determined via algebraic constraints of DAE (stage 2)



(a) Control input

(b) Consistent state variables

Fig. 3.4: Feedforward control: Specification of flat output $x_4(t)$ with $\Delta\vartheta = 10\,\mathrm{K}$



(a) Control input

(b) Consistent state variables

Fig. 3.5: Feedforward control: Specification of non-flat output $x_3\,(t)$ with $\Delta\vartheta = 10\,\mathrm{K}$

3.4.4 Structural Analysis for State and Disturbance Estimation

In analogy to the previous subsections dealing with the DAE-based design of feedforward control laws, the structural analysis is performed for the state and disturbance estimation task. In this case, an algebraic constraint $\mathbf{g}$ is defined which relates the temperatures x_1 and x_4 to the measured values $y_{m,1}$ and $y_{m,2}$. Calculating the corresponding Lie derivatives leads to the following scheme showing that $L^1_{\mathbf{f}}g_1 = 0, L^1_{\mathbf{f}}g_2 = 0, L^2_{\mathbf{f}}g_2 = 0$ can be solved for all unknown quantities:

	x_1	x_2	x_3	x_4	e	t	u
$\dot{x}_1$	◇	●			●		◇
$\dot{x}_2$	◇	●	●		●		
$\dot{x}_3$		●	●	◇	●		
$\dot{x}_4$			●	◇	●		
$g_1(\mathbf{x},t)$	◇					◇	
$g_2(\mathbf{x},t)$				◇		◇	

	x_1	x_2	x_3	x_4	e	t	u
$L^0_{\mathbf{f}}g_1$	◇					◇	
$L^1_{\mathbf{f}}g_1$	◇	●			●	◇	
$L^0_{\mathbf{f}}g_2$				◇		◇	
$L^1_{\mathbf{f}}g_2$			●	◇	●	◇	
$L^2_{\mathbf{f}}g_2$		●	●	◇	●	◇	◇

As an alternative to the solution summarized in this scheme, the second derivative of the first measured output corresponding to the constraint $L^2_{\mathbf{f}}g_1 = 0$ could be used instead of the constraint $L^2_{\mathbf{f}}g_2 = 0$. Note that the variation of the measured outputs is approximated by a linear function in time with the interval uncertainties given in Subsection 3.4.1.

The algorithm can be extended easily to higher order approximations of measured outputs, for example to arbitrary polynomial representations in t. Here, to reduce the sensitivity of the approximation of the output variables with respect to measurement noise and to avoid excessively wide intervals for the representation of approximation errors, verified least squares estimates can be used to determine the corresponding output representations g_1 and g_2, where the number of measured data is chosen significantly larger than the number of coefficients to be determined.

To visualize the changes that occur if a different finite volume representation of the heat transfer equation is used, the analysis of the system structure is repeated for $n = 5$, where the control input again only acts on the first rod segment. As in the previous case, the selection of the orders of the derivatives of the output variables is not unique. Instead of the third time derivative of the second output specified by $L_{\mathbf{f}}^3 g_2 = 0$, the second derivative of the first measured output corresponding to the constraint $L_{\mathbf{f}}^2 g_1 = 0$ can be used.

	x_1	x_2	x_3	x_4	x_5	e	t	u
$\dot{x}_1$	◇	●				●		◇
$\dot{x}_2$	◇	●	●			●		
$\dot{x}_3$		●	●	●		●		
$\dot{x}_4$			●	●	◇	●		
$\dot{x}_5$			●	●	◇	●		
$g_1(\mathbf{x},t)$	◇						◇	
$g_2(\mathbf{x},t)$					◇		◇	

	x_1	x_2	x_3	x_4	x_5	e	t	u
$L_{\mathbf{f}}^0 g_1$	◇						◇	
$L_{\mathbf{f}}^1 g_1$	◇	●				●	◇	◇
$L_{\mathbf{f}}^0 g_2$					◇		◇	
$L_{\mathbf{f}}^1 g_2$				●	◇	●	◇	◇
$L_{\mathbf{f}}^2 g_2$			●	●	◇	●	◇	◇
$L_{\mathbf{f}}^3 g_2$		●	●	●	◇	●	◇	◇

Legend:

◇	a-priori known from measurement or control design
●	determined via algebraic constraints of the DAE system

3.5 Dynamic Extensions for Feedforward Control Design

In this section, we demonstrate the basic procedure for a dynamic extension of system models for the design of exact feedforward control strategies using the example of an autonomous robot.

3.5.1 Example — Modeling of an Autonomous Robot

Consider the autonomous robot in Fig. 3.6. Its equations of motion on the $(x_1\ ;\ x_2)$–plane with the translational velocity u_1 and the angular velocity u_2 as inputs are given by the ODEs

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \cos(x_3(t)) \\ \sin(x_3(t)) \\ 0 \end{bmatrix} u_1(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u_2(t) \quad , \tag{3.42}$$

where x_3 denotes the angle of the orientation according to Fig. 3.6.

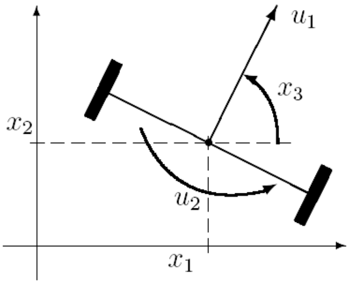


Fig. 3.6: Control of an autonomous robot

In the following, we consider the computation of feedforward control strategies for the inputs u_1 and u_2 such that the actual position of the robot is consistent with a predefined trajectory $\left(x_{1,d}\ ;\ x_{2,d}\right)$. Obviously, we have to assume consistent initial positions $x_1(0) = x_{1,d}(0)$ and $x_2(0) = x_{2,d}(0)$ for this task.

3.5.2 Feedforward Control Design

To determine dependencies of all system states $x_1(t)$, $x_2(t)$, $x_3(t)$ and all inputs $u_1(t)$, $u_2(t)$ on the desired trajectories $x_{1,d}(t)$, $x_{2,d}(t)$, both output equations $y_1 = x_1$ and $y_2 = x_2$ have to be differentiated twice. The relative degrees of the control input u_2 are equal to $\delta_1 = \delta_2 = 2$ in both cases according to

$$\begin{aligned} \dot{x}_1(t) &= \cos(x_3(t))\,u_1(t) \\ \ddot{x}_1(t) &= -\sin(x_3(t))\,u_2(t)\,u_1(t) + \cos(x_3(t))\,\dot{u}_1(t) \end{aligned} \tag{3.43}$$

and

$$\begin{aligned} \dot{x}_2(t) &= \sin(x_3(t))\,u_1(t) \\ \ddot{x}_2(t) &= \cos(x_3(t))\,u_2(t)\,u_1(t) + \sin(x_3(t))\,\dot{u}_1(t) \quad . \end{aligned} \tag{3.44}$$

Thus, for this system there are four constraints but only three unknowns (x_3, u_1, u_2). To solve this problem, we extend the dynamical system model (3.42) with an additional state variable for the input u_1 which appears as the the first derivative $\dot{u}_1$ in the Eqs. (3.43) and (3.44) for both $\ddot{x}_1$ and $\ddot{x}_2$. Thus, we obtain the new system model

$$\dot{z}(t) = \begin{bmatrix} \cos(z_3(t)) \\ \sin(z_3(t)) \\ 0 \\ 0 \end{bmatrix} z_4(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_1(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} v_2(t) \tag{3.45}$$

with the state variables $z_1 := x_1$, $z_2 := x_2$, $z_3 := x_3$, $z_4 := u_1$. The differentiation of the output equations now leads to

$$\begin{aligned} \dot{z}_1(t) &= \cos(z_3(t))\, z_4(t) \\ \ddot{z}_1(t) &= -\sin(z_3(t))\, z_4(t)\, v_2(t) + \cos(z_3(t))\, v_1(t) \end{aligned} \tag{3.46}$$

and

$$\begin{aligned} \dot{z}_2(t) &= \sin(z_3(t))\, z_4(t) \\ \ddot{z}_2(t) &= \cos(z_3(t))\, z_4(t)\, v_2(t) + \sin(z_3(t))\, v_1(t) \end{aligned} \tag{3.47}$$

with the new control inputs $v_1(t)$ and $v_2(t)$. These controls as well as the states $z_3(t)$ and $z_4(t)$ can be computed in the feedforward control design by substituting the desired trajectories $x_{1,d}(t)$ and $x_{2,d}(t)$ for $x_1(t)$ and $x_2(t)$, respectively.

This necessity for dynamic state extensions is typical for non-quasi-linear DAE systems and, generally, for differentially flat systems for which the sum of the relative degrees exceeds the dimension of the state vector. For the extended system (3.45), the feedforward control and state estimation procedures introduced in Section 3.2 can be applied.

The basis for the development of a general algorithm for automatic state extensions is the structural analysis performed by VALENCIA-IVP. For the original model of the autonomous robot, the corresponding result is:

	x_1	x_2	x_3	t	u_1	u_2
$\dot{x}_1$			●		●	
$\dot{x}_2$			●		●	
$\dot{x}_3$						●
$g_1(\mathbf{x},t)$	◇			◇		
$g_2(\mathbf{x},t)$		◇		◇		

	x_1	x_2	x_3	t	u_1	u_2
$L_{\mathbf{f}}^0 g_1$	◇			◇		
$L_{\mathbf{f}}^1 g_1$			●	◇	●	
$L_{\mathbf{f}}^0 g_2$		◇		◇		
$L_{\mathbf{f}}^1 g_2$			●	◇	●	

In this case, differentiation of the algebraic constraints is stopped as soon as one of the Lie derivatives depends on the algebraic state variable u_1 corresponding to the first control input. Since the resulting set of equations is underdetermined and, therefore, cannot be solved for u_2, the dynamic extension in the input u_1 is performed automatically. This extension leads to the following uniquely solvable system structure:

	z_1	z_2	z_3	z_4	t	v_1	v_2
$\dot{z}_1$			●	●			
$\dot{z}_2$			●	●			
$\dot{z}_3$							●
$\dot{z}_4$						●	
$g_1(\mathbf{z},t)$	◇				◇		
$g_2(\mathbf{z},t)$		◇			◇		

	z_1	z_2	z_3	z_4	t	v_1	v_2
$L_{\mathbf{f}}^0 g_1$	◇				◇		
$L_{\mathbf{f}}^1 g_1$			●	●	◇		
$L_{\mathbf{f}}^2 g_1$			●	●	◇	●	●
$L_{\mathbf{f}}^0 g_2$		◇			◇		
$L_{\mathbf{f}}^1 g_2$			●	●	◇		
$L_{\mathbf{f}}^2 g_2$			●	●	◇	●	●

Legend:

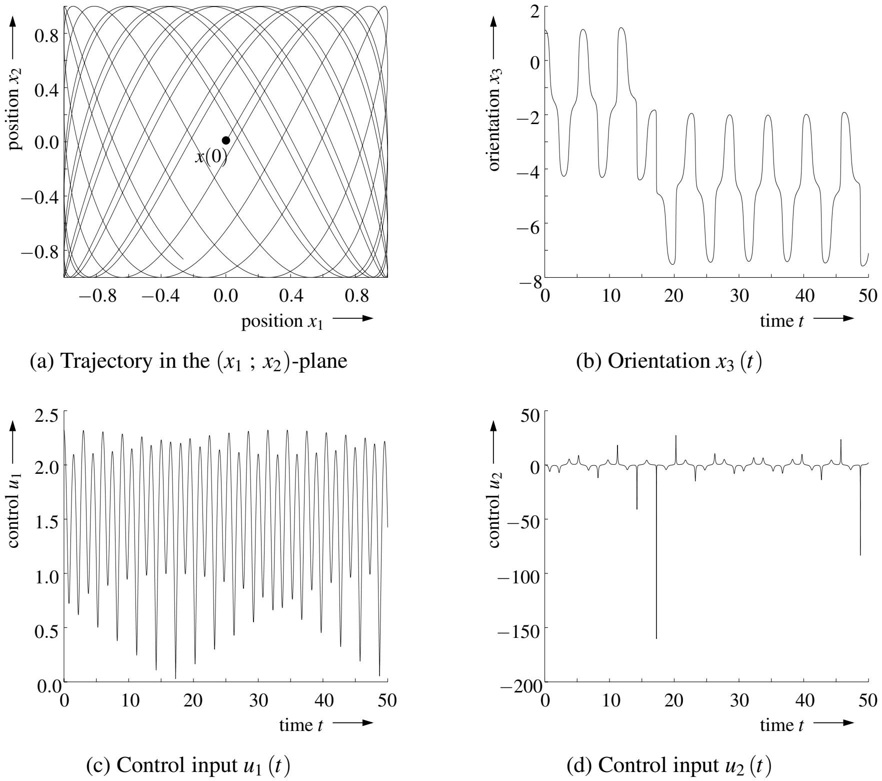
◇	a-priori known
●	determined via algebraic constraints of the DAE system

A typical result for the DAE-based feedforward control synthesis is shown in Fig. 3.7. Here, DAETS has been used as for the heat transfer problem in the previous section. In contrast to VALENCIA-IVP, the structural analysis included in DAETS asks the user to provide initial values for the complete state vector $\mathbf{x}(0)$ as well as for $u_1(0)$ and for the two derivatives $\dot{x}_1(0)$ and $\dot{x}_2(0)$ if the control task is formulated by the DAE problem consisting of the equations (3.42) and the constraints

$$\begin{aligned} 0 &= x_1(t) - x_{1,d}(t) = x_1(t) - \sin(t) \quad \text{and} \\ 0 &= x_2(t) - x_{2,d}(t) = x_2(t) - \sin\left(\frac{2}{3}\pi t\right) \quad . \end{aligned} \tag{3.48}$$

3.6 Conclusions and Outlook on Future Research

In this paper, interval-based approaches for the verification and implementation of robust control strategies were presented and applied to a finite volume representation of a distributed heating system as well as a model of an autonomous robot. For these systems, feedforward control laws which are computable online using VALENCIA-IVP were derived and extended by a classical output feedback for compensation of model and parameter uncertainties and neglected disturbances. Furthermore, a verified estimation procedure for internal system states and disturbances was described. It is implemented using a one-stage approach instead of the classical two-stage procedure usually employed by other interval observers. This observer can be applied to verify the admissibility and reliability of classical non-verified observers such as Luenberger-type observers by checking the guaranteed inclusion of the non-verified estimates in the corresponding interval bounds.



(a) Trajectory in the $(x_1\ ;\ x_2)$-plane

(b) Orientation $x_3\ (t)$

(c) Control input $u_1\ (t)$

(d) Control input $u_2\ (t)$

Fig. 3.7: Feedforward control for the autonomous robot using DAETS

In future work, we will investigate further relations between reachability and controllability of states and the solvability of DAEs describing feedforward control problems. Moreover, the routine implemented in ValEncIA-IVP for the detection of hidden algebraic constraints will be generalized. The goal will be to fully extend the presented automated feedforward control to multiple-input multiple-output systems for which desired output trajectories are prescribed for non-flat outputs and for which ambiguities in the solution might exist. One of the tasks will be a more detailed investigation of analogies and differences between the currently used structural analysis in ValEncIA-IVP and the one implemented in the non-verified DAE solver DAETS which employs the Dulmage-Mendelson algorithm to determine consistent initial conditions and consistent solutions to IVPs for DAEs.

Finally, combinations with verified tools for stability analysis based on interval evaluation of Lyapunov functions will be developed further to prove stability of non-observable or non-controllable internal dynamics and simultaneously to adapt controller structures to ensure asymptotically stable behavior.

References

1. Ackermann, J., Blue, P., Bünte, T., Güvenc, L., Kaesbauer, D., Kordt, M., Muhler, M., Odenthal, D.: Robust Control: The Parameter Space Approach, 2nd edn. Springer–Verlag, London (2002)
2. Aschemann, H., Minisini, J., Rauh, A.: Interval Arithmetic Techniques for the Design of Controllers for Nonlinear Dynamical Systems with Applications in Mechatronics — Part 1. Izvestiya RAN. Teoriya i sistemy upravleniya (Journal of Computer and Systems Sciences International) (3), 3–14 (2010)
3. Auer, E., Rauh, A., Hofer, E.P., Luther, W.: Validated Modeling of Mechanical Systems with SMARTMOBILE: Improvement of Performance by VALENCIA-IVP. In: Proc. of Dagstuhl Seminar 06021: Reliable Implementation of Real Number Algorithms: Theory and Practice, Lecture Notes in Computer Science, pp. 1–27 (2008)
4. Bendsten, C., Stauning, O.: FADBAD++, Version 2.1 (2007). http://www.fadbad.com
5. Berz, M., Makino, K.: COSY INFINITY Version 8.1. User's Guide and Reference Manual. Tech. Rep. MSU HEP 20704, Michigan State University (2002)
6. Delanoue, N.: Algoritmes numériques pour l'analyse topologique — Analyse par intervalles et théorie des graphes. Ph.D. thesis, École Doctorale d'Angers (2006). In French
7. Fliess, M., Lévine, J., Martin, P., Rouchon, P.: Flatness and Defect of Nonlinear Systems: Introductory Theory and Examples. International Journal of Control **61**, 1327–1361 (1995)
8. Freihold, M., Hofer, E.P.: Derivation of Physically Motivated Constraints For Efficient Interval Simulations Applied to the Analysis of Uncertain Dynamical Systems. Special Issue of the International Journal of Applied Mathematics and Computer Science AMCS, "Verified Methods: Applications in Medicine and Engineering" **19**(3), 485–499 (2009)
9. Griewank, A., Walther, A.: Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation. SIAM, Philadelphia (2008)
10. Isidori, A.: Nonlinear Control Systems 1. An Introduction. Springer–Verlag, Berlin (1995)
11. Jaulin, L., Kieffer, M., Didrit, O., Walter, É.: Applied Interval Analysis. Springer–Verlag, London (2001)
12. Keil, C.: PROFIL/BIAS, Version 2.0.8 (2008). www.ti3.tu-harburg.de/keil/profil/
13. Kletting, M., Rauh, A., Aschemann, H., Hofer, E.P.: Interval Observer Design Based on Taylor Models for Nonlinear Uncertain Continuous-Time Systems. In: CD-Proc. of the 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006. IEEE Computer Society, Duisburg, Germany (2007)
14. Krawczyk, R.: Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken. Computing **4**, 189–201 (1969). In German
15. Lerch, M., Tischler, G., Wolff von Gudenberg, J., Hofschuster, W., Krämer, W.: The Interval Library `filib++ 2.0` : Design, Features and Sample Programs. Tech. Rep. 2001/4, Bergische Universität GH Wuppertal (2001)
16. Lin, Y., Stadtherr, M.: Validated solution of initial value problems for ODEs with interval parameters. In: NSF Workshop Proceeding on Reliable Engineering Computing. Savannah GA (2006)
17. Marquez, H.J.: Nonlinear Control Systems. John Wiley & Sons, Inc., New Jersey (2003)
18. Moore, R.: Interval Arithmetic. Prentice-Hall, Englewood Cliffs, New Jersey (1966)
19. Nedialkov, N.S.: Interval Tools for ODEs and DAEs. In: CD-Proc. of the 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006. IEEE Computer Society, Duisburg, Germany (2007)
20. Nedialkov, N.S., Pryce, J.D.: Solving Differential-Algebraic Equations by Taylor Series (I): Computing Taylor Coefficients. BIT **45**(3), 561–591 (2005)
21. Nedialkov, N.S., Pryce, J.D.: Solving Differential-Algebraic Equations by Taylor Series (II): Computing the System Jacobian. BIT **47**(1), 121–135 (2007)

22. Nedialkov, N.S., Pryce, J.D.: DAETS — Differential-Algebraic Equations by Taylor Series (2008). http://www.cas.mcmaster.ca/~nedialk/daets/
23. Nedialkov, N.S., Pryce, J.D.: Solving Differential-Algebraic Equations by Taylor Series (III): the DAETS Code. J. Numerical Analysis, Industrial and Applied Mathematics **3**, 61–80 (2008)
24. Rauh, A., Aschemann, H.: Sensitivity-Based Feedforward and Feedback Control Using Algorithmic Differentiation. In: Proc. of IEEE Intl. Conference on Methods and Models in Automation and Robotics MMAR 2010. Miedzyzdroje, Poland (2010)
25. Rauh, A., Kersten, J., Auer, E., Aschemann, H.: Sensitivity Analysis for Reliable Feedforward and Feedback Control of Dynamical Systems with Uncertainties. In: 8th Intl. Conference on Structural Dynamics EURODYN 2011. Leuven, Belgium (2011)
26. Rauh, A., Auer, E., Freihold, M., Hofer, E.P., Aschemann, H.: Detection and Reduction of Overestimation in Guaranteed Simulations of Hamiltonian Systems. In: Proc. of the 13th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2008 in Special Issue of Reliable Computing. El Paso, Texas, USA (2010)
27. Rauh, A., Brill, M., Günther, C.: A Novel Interval Arithmetic Approach for Solving Differential-Algebraic Equations with VALENCIA-IVP. Special Issue of the International Journal of Applied Mathematics and Computer Science AMCS, "Verified Methods: Applications in Medicine and Engineering" **19**(3), 381–397 (2009)
28. Rauh, A., Grigoryev, V., Aschemann, H., Paschen, M.: Incremental Gain Scheduling and Sensitivity-Based Control for Underactuated Ships. In: Proc. of IFAC Conference on Control Applications in Marine Systems, CAMS 2010. Rostock-Warnemünde, Germany (2010)
29. Rauh, A., Hofer, E.P.: Interval Methods for Optimal Control. In: A. Frediani, G. Buttazzo (eds.) Proc. of the 47th Workshop on Variational Analysis and Aerospace Engineering 2007, pp. 397–418. Springer–Verlag, School of Mathematics, Erice, Italy (2009)
30. Rauh, A., Minisini, J., Aschemann, H.: Interval Arithmetic Techniques for the Design of Controllers for Nonlinear Dynamical Systems with Applications in Mechatronics — Part 2. Izvestiya RAN. Teoriya i sistemy upravleniya (Journal of Computer and Systems Sciences International) (2010). Under review
31. Rauh, A., Minisini, J., Hofer, E.P.: Towards the Development of an Interval Arithmetic Environment for Validated Computer-Aided Design and Verification of Systems in Control Engineering. In: Proc. of Dagstuhl Seminar 08021: Numerical Validation in Current Hardware Architectures, *Lecture Notes in Computer Science*, vol. 5492, pp. 175–188. Springer–Verlag, Dagstuhl, Germany (2008)
32. Rauh, A., Minisini, J., Hofer, E.P.: Verification Techniques for Sensitivity Analysis and Design of Controllers for Nonlinear Dynamic Systems with Uncertainties. Special Issue of the International Journal of Applied Mathematics and Computer Science AMCS, "Verified Methods: Applications in Medicine and Engineering" **19**(3), 425–439 (2009)
33. Rauh, A., Minisini, J., Hofer, E.P., Aschemann, H.: Robust and Optimal Control of Uncertain Dynamical Systems with State-Dependent Switchings Using Interval Arithmetic. In: Proc. of the 13th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2008 in Special Issue of Reliable Computing. El Paso, Texas, USA (2010)

Chapter 4
Analyzing Reachability of Linear Dynamic Systems with Parametric Uncertainties

Matthias Althoff (✉), Bruce H. Krogh, and Olaf Stursberg

Abstract As an important approach to analyzing safety of a dynamic system, this paper considers the task of computing overapproximations of reachable sets, i.e. the set of states which is reachable from a given initial set of states. The class of systems under investigation are linear, time-invariant systems with parametric uncertainties and uncertain but bounded input. The possible set of system matrices due to uncertain parameters is represented by matrix zonotopes and interval matrices – computational techniques for both representations are presented. The reachable set is represented by zonotopes, which makes it possible to apply the approach to systems of 100 continuous state variables with computation times of a few minutes. This is demonstrated for randomized examples as well as a transmission line example.

4.1 Introduction

Reachability analysis deals with the problem of finding the set of states that a system can reach when starting from a specified set of initial states in finite or infinite time. One of the main purposes of reachability analysis is to demonstrate the safe execution of a system by proving that the system does not reach any unsafe state. This

Matthias Althoff
Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, USA
e-mail: malthoff@ece.cmu.edu

Bruce H. Krogh
Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, USA
e-mail: krogh@ece.cmu.edu

Olaf Stursberg
University of Kassel, Control and System Theory (FB16), Wilhelmshöher Allee 73,
D-34121 Kassel, Germany
e-mail: stursberg@uni-kassel.de

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_4,

is illustrated for a two-dimensional example with states x_1, x_2 in Fig. 4.1. Besides the safety verification problem, reachability analysis is a useful tool for robustness analysis [1], abstraction of hybrid systems [2], and state-bounding observers [3].

In this work, an efficient algorithm for computing reachable sets of continuous-time linear systems with uncertain inputs/disturbances and constant but uncertain parameters is presented. One advantage of the proposed method is that the computational complexity is moderate in terms of the system dimension. As shown by earlier work, the reachability algorithm for linear systems can be extended to the analysis of nonlinear systems [4] and hybrid systems [5]. Thus, the reachability analysis of linear systems can be seen as a basic module for the reachability analysis of more complicated system classes.

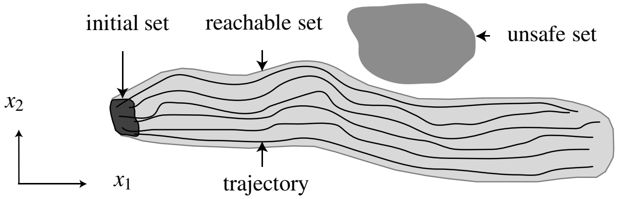


Fig. 4.1: An empty intersection of (an overapproximation of) the reachable set with an unsafe set of states verifies system safety

For systems with derivative bounds $\dot{\mathbf{x}} \in P$, where $\mathbf{x} \in \mathbb{R}^n$ and P is a bounded convex polyhedron (polytope) in $\mathbb{R}^n$, the reachable set can be represented by polyhedra [6]. Reachable sets of such systems can be used as a basis for the reachability analysis of linear or even more complex systems, such as nonlinear and hybrid systems [7, 8].

Other work deals directly with linear systems $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{u}(t)$, where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in U \subset \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{n \times n}$. Exact reachable sets of linear systems can only be obtained in special cases; in general one has to compute overapproximations to perform system verification [9]. Approaches to this class of systems can be classified by the geometric representation used for the reachable sets: polytopes [10], ellipsoids [11], oriented rectangular hulls [12], zonotopes [13, 14], or level sets [15]. Support functions [16] unify these methods, except of the use of level sets. If uncertain parameters are considered, most existing algorithms are based on interval methods and multidimensional intervals (hyperrectangles) to represent reachable sets [17–19]. Similar techniques are used for validated integration methods of ordinary differential equations, which are typically applied to smaller uncertainties in the initial states [20–22].

Besides the mentioned techniques that are based on guaranteed set integration, for which an overview can be found in [23], one can verify the safety of a system with barrier certificates [24] or simulation based techniques, e.g. [25, 26].

Previous work addressed the computation of reachable sets of linear systems with uncertain parameters [27]. Recently, this approach has been extended to linear systems with time-varying uncertain parameters [28]. In these works, the reachable sets are represented by zonotopes, which offer a more general representation compared to multidimensional intervals, which are typically used for this class of problems. Zonotopes are also a more efficient alternative to arbitrary polytopes for reachability analysis of linear systems [14]. The novelties for the follow-up work presented here are:

- Improved computational techniques: Dependencies between the elements of state transition matrices due to common parameters are considered when computing with matrix zonotopes.
- A norm bound for the computation of matrix exponential sets is derived.
- Performance evaluations of methods for computing matrix exponential sets are conducted.
- Properties of a new transmission line example are verified.

This book chapter is organized as follows. In Section 4.2, the problem of computing reachable sets is introduced, and a brief description of the used algorithmic procedure is given. The formulas for computing reachable sets of linear systems under uncertain initial states, parameters, and inputs are derived in Section 4.3. These formulas are based on the set of possible state transition matrices, of which the computation is described in Section 4.4. The usefulness of the presented approach is demonstrated for a transmission line example, and randomly generated examples in Section 4.5.

4.2 Problem Formulation

We consider time-invariant linear systems of the form

$$\dot{\mathbf{x}}(t) = \mathbf{A}\,\mathbf{x}(t) + \mathbf{u}(t), \quad \mathbf{A} \in \mathscr{A}, \mathbf{u}(\cdot) \in \mathscr{U}_{[0,t_f]}, \mathbf{x}(0) \in \mathscr{X}_0, t \in [0,t_f],$$

where $\mathbf{u}(t) : \mathbb{R}^+ \to \mathbb{R}^n$ is an input function over time, $\mathscr{A}$ is the set of system matrices $\mathbf{A}$, $\mathscr{X}_0$ is the set of initial states, and $t_f \in \mathbb{R}^+$ is the time horizon. The set of input functions is defined as $\mathscr{U}_{[0,t_f]} = \{\mathbf{u}(\cdot)|\mathbf{u}(\cdot) \text{ is piecewise continuous}, \mathbf{u}(t) \in \mathscr{U}, t \in [0,t_f]\}$, where $\mathscr{U}$ is the set of possible input values. The notation $\mathbf{u}(\cdot)$ refers to trajectories rather than the explicit value at time t. Note that the commonly used input formulation $\mathbf{B}\,\tilde{\mathbf{u}}(t)$ is included in $\mathbf{u}(t)$ when defining $\mathscr{U} = \{\mathbf{B}\tilde{\mathbf{u}}|\tilde{\mathbf{u}} \in \tilde{\mathscr{U}}\}$.

The objective of this work is to compute the set of reachable states

$$\mathscr{R}^e([0,t_f]) = \Big\{\mathbf{x}\Big|\mathbf{x} = \int_0^t (\mathbf{A}\,\mathbf{x}(\tau) + \mathbf{u}(\tau))d\tau, \mathbf{A} \in \mathscr{A}, \\ \mathbf{u}(\cdot) \in \mathscr{U}_{[0,t_f]}, \mathbf{x}(0) \in \mathscr{X}_0, t \in [0,t_f]\Big\}.$$

The fact that $\mathscr{R}^e([0,t_f])$ refers to the exact reachable set is indicated by the superscript e. However, the reachable set for uncertain time-invariant linear systems cannot be computed exactly for arbitrary $\mathbf{A}$ and $\mathbf{u}(\cdot)$ [9]. Therefore, overapproximations $\mathscr{R}([0,t_f]) \supseteq \mathscr{R}^e([0,t_f])$ are computed in this work. The task is to find algorithms that bound the overapproximation as tightly as possible, while at the same time ensuring that the algorithms are efficient and scale well with the system dimension n. Ensuring tightness of the enclosure is a challenging task due to the wrapping effect, which is understood as the propagation of overapproximations through the computations of successive time steps [29].

The basic principle of many reachability algorithms, including the approach presented here, is to compute the reachable set for consecutive time intervals $\mathscr{R}([t_{k-1},t_k])$, where $t_k = k \cdot r$ and $k \in \mathbb{N}$ is the time step; see [10,12,14,30]. The complete reachable set is then obtained by: $\mathscr{R}([0,t_f]) = \bigcup_{k=1\ldots t_f/r} \mathscr{R}([t_{k-1},t_k])$, where t_f is a multiple of r. Since the union is represented as a list of the sets $\mathscr{R}([t_{k-1},t_k])$, the focus of this work is on the computation of a single time interval $[0,r]$. The basic steps for the computation of $\mathscr{R}([0,r])$ are shown in Fig. 4.2 and are summarized as follows:

1. Computation of the reachable set $\mathscr{H}(r)$ without the input (homogeneous solution), but with consideration of the set $\mathscr{A}$ of system matrices;
2. Generation of the convex hull of the solution at $t = r$ and the initial set;
3. Enlargement of the convex hull to ensure enclosure of all trajectories for the time interval $t \in [0,r]$. The enlargement compensates for two assumptions made in steps 1 and 2: The first assumption was that the system has no input. The second one was that trajectories between the initial set and the reachable set $\mathscr{H}(r)$ are straight lines for which the convex hull computation would be sufficient.

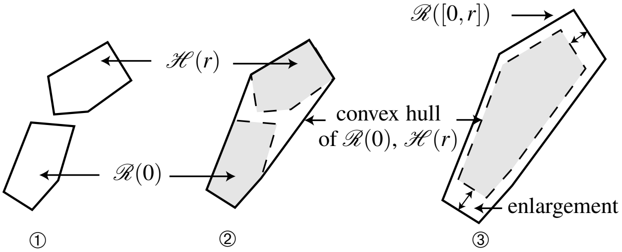


Fig. 4.2: Computation of the reachable set for a time interval

It is guaranteed that the formulas derived below return reachable sets that enclose all possible trajectories. The implementation of the algorithms in this work neglects the effect of floating-point errors caused by the finite number of stored digits in computers. This effect can be taken care of by exchanging floating-point arithmetic by interval arithmetic [31], which propagates the rounding errors.

4.3 Overapproximating the Reachable Set

It is well known that the solution of an autonomous linear time-invariant system ($\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x}$) is provided by the state transition matrix: $\mathbf{x}(t) = \boldsymbol{\Phi}(t,t_0)\mathbf{x}_0$, where $\boldsymbol{\Phi}(t,t_0) = e^{\mathbf{A}(t-t_0)}$. When the initial state is uncertain within $\mathscr{R}(t_0)$, the set of reachable states at time t is $\mathscr{R}(t) = \{e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 | \mathbf{x}_0 \in \mathscr{R}(t_0)\}$. If additionally, the system matrix is uncertain, one has to compute the reachable set as $\mathscr{R}(t) = \{e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 | \mathbf{A} \in \mathscr{A}, \mathbf{x}_0 \in \mathscr{R}(t_0)\}$. The computation of the set of possible state transition matrices is discussed first. Then, the extensions for reachable sets of time intervals $[0,r]$ and under the influence of uncertain inputs are presented. Without loss of generality, it is assumed that $t_0 = 0$ from now on, so that $\boldsymbol{\Phi}(t) = \boldsymbol{\Phi}(t,t_0)$.

4.3.1 Overapproximating the State Transition Matrix

In order to make the computation of the set of state transition matrices $\{e^{\mathbf{A}t} | \mathbf{A} \in \mathscr{A}\}$ tractable for matrix zonotopes and interval matrices, some set-based operations have to be computed independently. The set computations that remain dependent are indicated by a special notation. Letting $\circ$ denote either addition or multiplication, then the exact evaluation is denoted by

$$[\![\mathbf{A} \circ \mathbf{A}]\!]_{\mathbf{A}\in\mathscr{A}} := \{\mathbf{A} \circ \mathbf{A} | \mathbf{A} \in \mathscr{A}\}, \tag{4.1}$$

while an independent evaluation is denoted by

$$\mathscr{A} \circ \mathscr{A} := \{\mathbf{A}_1 \circ \mathbf{A}_2 | \mathbf{A}_1 \in \mathscr{A}, \mathbf{A}_2 \in \mathscr{A}\}.$$

Using an independent evaluation of operands, one obtains an overapproximation in general, e.g.

$$[\![(\mathbf{A}+\mathbf{B})\mathbf{C}]\!]_{\substack{\mathbf{A}\in\mathscr{A}\\ \mathbf{B}\in\mathscr{B}\\ \mathbf{C}\in\mathscr{C}}} \subseteq [\![\mathbf{AC}]\!]_{\substack{\mathbf{A}\in\mathscr{A}\\ \mathbf{C}\in\mathscr{C}}} + [\![\mathbf{BC}]\!]_{\substack{\mathbf{B}\in\mathscr{B}\\ \mathbf{C}\in\mathscr{C}}} = \mathscr{A}\mathscr{C} + \mathscr{B}\mathscr{C}.$$

The notation introduced above makes it possible to formulate an overapproximation of the set of matrices $\overline{\mathscr{M}}(t) := [\![e^{\mathbf{A}t}]\!]_{\mathbf{A}\in\mathscr{A}}$ based on the Taylor series of $e^{\mathbf{A}t}$. For typical step sizes in time used in reachability analysis, only the first terms of the Taylor series contribute significantly to the solution. Thus, the dependent set-based evaluation is performed up to second order, while higher order terms are evaluated independently; that is,

$$\begin{aligned}\overline{\mathscr{M}}(t) &= \left[\!\!\left[I + \mathbf{A}t + \frac{1}{2!}(\mathbf{A}t)^2 + \frac{1}{3!}(\mathbf{A}t)^3 + \frac{1}{4!}(\mathbf{A}t)^4 + \ldots \right]\!\!\right]_{\mathbf{A}\in\mathscr{A}} \\ &\subseteq \left[\!\!\left[I + \mathbf{A}t + \frac{1}{2!}(\mathbf{A}t)^2 \right]\!\!\right]_{\mathbf{A}\in\mathscr{A}} + \frac{1}{3!}(\mathscr{A}t)^3 + \frac{1}{4!}(\mathscr{A}t)^4 + \ldots .\end{aligned} \tag{4.2}$$

It is shown below that the computation above is always bounded when the set of matrix values $\mathscr{A}$ and time t is bounded. Thereto, the norm of a set of matrices is defined as

$$||\mathscr{A}|| = \sup\{||\mathbf{A}|| \,|\mathbf{A} \in \mathscr{A}\}, \tag{4.3}$$

where $||\mathbf{A}||$ denotes an arbitrary matrix norm, while special norms, such as the infinity norm, will be denoted by $||\mathscr{A}||_\infty$. Applying the matrix norm, one obtains

$$||[\![e^{\mathbf{A}t}]\!]_{\mathbf{A}\in\mathscr{A}}|| \leq \sum_{i=0}^{\infty} \frac{1}{i!} ||\mathscr{A}||^i t^i = e^{||\mathscr{A}||t},$$

which is bounded for $||\mathscr{A}|| < \infty$ and $t < \infty$.

In order to compute $[\![e^{\mathbf{A}t}]\!]_{\mathbf{A}\in\mathscr{A}}$, the infinite sum in (4.2) has to be replaced by a finite sum to which a set of remainder matrices is added. The number of terms retained in the Taylor series is denoted by η.

Proposition 4.1 (State Transition Matrix Remainder). *The set of remainder matrices $\sum_{i=\eta+1}^{\infty} \frac{1}{i!}\mathscr{A}^i t^i$ is overapproximatedfor $|\mathscr{A}| \leq \mathbf{C} \in \mathbb{R}^{n\times n}$ by the interval matrix*

$$\mathscr{E}_{[i]}(t) = [-\mathbf{Y}(t), \mathbf{Y}(t)], \quad \mathbf{Y}(t) = e^{\mathbf{C}t} - \sum_{i=0}^{\eta} \frac{\mathbf{C}^i t^i}{i!}.$$

The absolute value of a matrix set is defined as the matrix in which each element is equal to the supremum of the absolute value of the corresponding element in each matrix in $\mathscr{A}$. That is, $|\mathscr{A}|_{i,j} = \sup\{|a_{i,j}|\,|\mathbf{A} \in \mathscr{A}\}$.

Proof. The multiplication of two matrix sets $\mathscr{A}$ and $\mathscr{B}$, where C and D are chosen such that $|\mathscr{A}| \leq \mathbf{C} \in \mathbb{R}^{n\times n}$ and $|\mathscr{B}| \leq \mathbf{D} \in \mathbb{R}^{n\times n}$, has the absolute value bound $|\mathscr{A}\,\mathscr{B}| \leq |\mathscr{A}|\,|\mathscr{B}| \leq CD$. From this it follows that $|\mathscr{A}^n| \leq \mathbf{C}^n$ such that

$$\left|\sum_{i=\eta+1}^{\infty} \frac{\mathscr{A}^i t^i}{i!}\right| \leq \sum_{i=\eta+1}^{\infty} \frac{|\mathscr{A}^i| t^i}{i!} \leq \sum_{i=\eta+1}^{\infty} \frac{\mathbf{C}^i t^i}{i!} = e^{\mathbf{C}t} - \sum_{i=0}^{\eta} \frac{\mathbf{C}^i t^i}{i!}. \qquad \square$$

Besides the presented Taylor method, there is a number of different techniques to compute the matrix exponential [32]. Unfortunately, these alternative approaches are not suitable for computations with matrix sets or do not provide error bounds. No error bounds can be provided when applying techniques which use solvers of ordinary differential equations [32]. Polynomial methods make it possible to obtain the matrix exponential from a finite sum $e^{At} = \sum_{i=0}^{n-1} \alpha_i(t) A^i$, where $\alpha_i(t)$ is a polynomial. However, the error introduced by the Taylor series remainder, which would be omitted using this technique, is small compared to the computation of the powers $\mathscr{A}^i$. Matrix decomposition methods, where $A = SBS^{-1}$ so that $e^{At} = Se^{Bt}S^{-1}$ suffer from the problem that the inverse of an uncertain matrix is hard to compute [33] and that for many techniques S is hard to obtain when A is uncertain, e.g. when S is a matrix of eigenvectors [34]. Splitting techniques which are based on the formula $e^{B+C} = \lim_{m\to\infty}(e^{B/m}e^{C/m})^m$ are not appropriate, too, since high powers of matrix sets are hard to compute.

4.3.2 Reachable Sets of Time Intervals

Given the homogeneous solution $\mathbf{x}_h(r) \in \overline{\mathscr{M}}(r)\mathbf{x}(0)$, the following approximation for the solution at intermediate points in time is suggested:

$$\hat{\mathbf{x}}_h(t) = \mathbf{x}(0) + \frac{t}{r}(\mathbf{M}\mathbf{x}(0) - \mathbf{x}(0)), \quad \mathbf{M} \in \overline{\mathscr{M}}(r), t \in [0,r]. \tag{4.4}$$

The error $\mathbf{x}_h(t) - \hat{\mathbf{x}}_h(t)$ made when applying this approximation is bounded by the set $\mathscr{F}(r)\mathbf{x}(0)$, where $\mathscr{F}(r)$ is a set of matrices such that $\mathbf{x}_h(t) \in \hat{\mathbf{x}}_h(t) + \mathscr{F}(r)\mathbf{x}(0)$. Using the inclusion $\mathbf{x}_h(t) \in \overline{\mathscr{M}}(t)\mathbf{x}(0)$ and replacing $\overline{\mathscr{M}}(t)$ by its Taylor series yields a formula for computing the set of matrices $\mathscr{F}$:

$$\begin{aligned}\mathscr{F}(r) &\supseteq \left\{ \sum_{i=0}^{\eta} \frac{\mathbf{A}_i t^i}{i!} + \mathscr{E}_{[i]}(t) - \mathbf{I} - \frac{t}{r}\Big(\sum_{i=0}^{\infty} \frac{\mathbf{A}_i r^i}{i!} + \mathscr{E}_{[i]}(r) - \mathbf{I}\Big) \middle| \mathbf{A}_i \in \mathscr{A}^i, t \in [0,r] \right\} \\ &= \left\{ \sum_{i=2}^{\eta} \frac{\mathscr{A}^i}{i!}(t^i - tr^{i-1}) + \mathscr{E}_{[i]}(t) - \frac{t}{r}\mathscr{E}_{[i]}(r) \middle| t \in [0,r] \right\}.\end{aligned}$$

In [27] it is shown that

$$[\varphi](i,r) := \left\{ t^i - tr^{i-1} \middle| t \in [0,r] \right\} = [(i^{\frac{-i}{i-1}} - i^{\frac{-1}{i-1}})r^i, 0].$$

It remains to compute $\mathscr{E}_{[i]}(t) - \frac{t}{r}\mathscr{E}_{[i]}(r)$. The matrix set $\mathscr{E}_{[i]}(t)$ is strictly increasing with time so that $\mathscr{E}_{[i]}(t) \in [0,1]\mathscr{E}_{[i]}(r)$ for $t \in [0,r]$. Thus,

$$\left\{ \mathscr{E}_{[i]}(t) - \frac{t}{r}\mathscr{E}_{[i]}(r) \middle| t \in [0,r] \right\} \subseteq \left\{ (\mu_1 - \mu_2)\mathscr{E}_{[i]}(r) \middle| \mu_1, \mu_2 \in [0,1] \right\} = [-1,1]\mathscr{E}_{[i]}(r)$$

and $[-1,1]\mathscr{E}_{[i]}(r) = \mathscr{E}_{[i]}(r)$ because $\mathscr{E}_{[i]}(t)$ has symmetric bounds. These simplifications make it possible to compute $\mathscr{F}(r)$ as

$$\mathscr{F}(r) = \sum_{i=2}^{\eta} \frac{\mathscr{A}^i}{i!}[\varphi](i,r) + \mathscr{E}_{[i]}(r).$$

Since all possible solutions of (4.4) are contained in the convex hull $\mathtt{CH}(\mathscr{R}(0) \cup \overline{\mathscr{M}}(r)\mathscr{R}(0))$, the reachable set for a time interval without input can be computed as $\mathscr{R}([0,r]) = \mathtt{CH}(\mathscr{R}(0) \cup \overline{\mathscr{M}}(r)\mathscr{R}(0)) + \mathscr{F}(r)\mathscr{R}(0)$.

4.3.3 Reachable Set of the Complete System

We now consider the additional contribution to the reachable set due to uncertain inputs. Since the superposition principle for linear systems can be applied, the reachable set of the input solution can be computed independently of the homogeneous solution. The input solution $\mathbf{x}_p(t)$ is bounded according to [35, Chap. 3] by

$$\mathbf{x}_p(t) \in \int_{t_0}^{t} \overline{\mathscr{M}}(t-\tau)\mathbf{u}(\tau)d\tau, \quad t \geq t_0. \tag{4.5}$$

In order to compute the reachable set due to uncertain inputs, the following proposition on distributivity of positive scalars and convex matrix sets is required.

Proposition 4.2 (Distributivity of Matrix Sets). *When $\mathscr{A}$ is convex and $a,b \in \mathbb{R}^+$:*

$$a\mathscr{A} + b\mathscr{A} = (a+b)\mathscr{A}.$$

Proof. It is always true that $(a+b)\mathscr{A} \subseteq a\mathscr{A} + b\mathscr{A}$, even if $\mathscr{A}$ is not convex. Further, due to the convexity it follows for the real-valued and arbitrary matrices $\mathbf{X}_1, \mathbf{X}_2 \in \mathscr{A}$ and the scalar $\alpha \in [0,1]$ that $\alpha\mathbf{X}_1 + (1-\alpha)\mathbf{X}_2 \in \mathscr{A}$. Making use of $a,b \geq 0$ this can be rewritten by choosing $\alpha = \frac{a}{a+b}$:

$$\frac{a}{a+b}\mathbf{X}_1 + \frac{b}{a+b}\mathbf{X}_2 \in \mathscr{A}$$

so that $a\mathbf{X}_1 + b\mathbf{X}_2 \in (a+b)\mathscr{A}$ and consequently $a\mathscr{A} + b\mathscr{A} \subseteq (a+b)\mathscr{A}$. □

Theorem 4.1 (Input Solution). *The set of reachable states due to the uncertain input $\mathbf{u}(t) \in \mathscr{U}$ is overapproximated as*

$$\mathscr{P}(t) = \sum_{i=0}^{\eta} \left(\frac{\mathtt{CH}(\mathscr{A}^i \mathscr{U}) t^{i+1}}{(i+1)!} \right) + \mathscr{E}_{[i]}(t)\, t\, |\mathscr{U}|.$$

Proof. The integral in (4.5) is solved for set-valued inputs by splitting the integral from t_0 to t into subintervals $[t_k, t_{k+1}]$, where $k \in \{0, \ldots, m-1\}$. For now, it is assumed that the input value taken from $\mathscr{U}$ is constant within time intervals $[t_k, t_{k+1}]$, so that $\mathscr{U}$ can be excluded from the integration:

$$\mathbf{x}_p(t) \in \sum_{k=0}^{m-1} \int_{t_k}^{t_{k+1}} \overline{\mathscr{M}}(t-\tau)d\tau\,\mathscr{U}. \tag{4.6}$$

This assumption will be overruled when choosing $m \to \infty$ later. Next, $\overline{\mathscr{M}}(t-\tau) = \sum_{i=0}^{\eta} \mathscr{A}^i (t-\tau)^i / i! + \mathscr{E}_{[i]}(t-\tau)$ is inserted so that

$$\int_{t_k}^{t_{k+1}} \overline{\mathscr{M}}(t-\tau)d\tau = \sum_{i=0}^{\eta} \frac{\mathscr{A}^i}{i!} \underbrace{\int_{t_k}^{t_{k+1}} (t-\tau)^i d\tau}_{=\int_{t-t_{k+1}}^{t-t_k} \tau^i d\tau} + \underbrace{\int_{t_k}^{t_{k+1}} \mathscr{E}_{[i]}(t-\tau)d\tau}_{=\int_{t-t_{k+1}}^{t-t_k} \mathscr{E}_{[i]}(\tau)d\tau} \tag{4.7}$$

The integral in (4.7) can be moved inside since the matrix values within $\mathscr{A}$ are not time-varying. Inserting (4.7) into (4.6) yields

$$\mathbf{x}_p(t) \in \sum_{k=0}^{m-1} \left(\sum_{i=0}^{\eta} \frac{\mathscr{A}^i}{i!} \int_{t-t_{k+1}}^{t-t_k} \tau^i d\tau + \int_{t-t_{k+1}}^{t-t_k} \mathscr{E}_{[i]}(\tau)d\tau \right) \mathscr{U}$$

Using $\sum_{k=0}^{m-1}\int_{t-t_{k+1}}^{t-t_k}\mathscr{E}_{[i]}(\tau)d\tau|\mathscr{U}| = \int_0^t \mathscr{E}_{[i]}(\tau)d\tau|\mathscr{U}|$, where $|\mathscr{U}|$ returns an axis-aligned box, and applying Prop. 4.2 yields

$$\mathbf{x}_p(t)\in\sum_{i=0}^{\eta}\frac{\mathtt{CH}(\mathscr{A}^i\mathscr{U})}{i!}\underbrace{\int_0^t\tau^i d\tau}_{=t^{i+1}/(i+1)}+\int_0^t\mathscr{E}_{[i]}(\tau)d\tau|\mathscr{U}|.$$

One can see that the result is independent of the number m of intermediate time intervals due to Prop. 4.2. This means that choosing $m\rightarrow\infty$ returns the same result so that the assumption of constant input values within time intervals can be overruled. It remains to compute the integral $[-\tilde{\mathbf{Y}}(t),\tilde{\mathbf{Y}}(t)] := \int_0^t\mathscr{E}_{[i]}(\tau)d\tau$, where

$$\tilde{\mathbf{Y}}(t)=\sum_{i=\eta+1}^{\infty}\frac{\mathbf{C}^i}{(i+1)!}t^{i+1}<\sum_{i=\eta+1}^{\infty}\frac{\mathbf{C}^i}{i!}t^{i+1}=\mathbf{Y}(t)t,$$

so that $\int_0^t\mathscr{E}_{[i]}(\tau)d\tau\subset\mathscr{E}_{[i]}(t)t$ and $\mathbf{Y}(t)$ is as introduced in Prop. 4.1. □

If the origin is contained in the set of possible inputs ($0\in\mathscr{U}$), it holds that $\mathscr{P}([0,r])=\mathscr{P}(r)$; see [27]. If this is not the case, some minor extensions are required [27]. Assuming that $0\in\mathscr{U}$, the overall algorithm for computing the reachable set can be stated in Algorithm 1.

Algorithm 1 Compute $\mathscr{R}([0,t_f])$

```
Input: Initial set $\mathscr{R}(0)$, set of state transition matrices $\overline{\mathscr{M}}(r)$, input set $\mathscr{U}$, set of correction matrices $\mathscr{F}(r)$, time increment $r$, time horizon $t_f$
Output: $\mathscr{R}([0,t_f])$

$\mathscr{H}_0 = \mathtt{CH}(\mathscr{R}(0)\cup\overline{\mathscr{M}}(r)\mathscr{R}(0))+\mathscr{F}(r)\mathscr{R}(0)$
$\mathscr{P}_0 = \sum_{i=0}^{\eta}\left(\frac{\mathtt{CH}(\mathscr{A}^i\mathscr{U})r^{i+1}}{(i+1)!}\right)+\mathscr{E}_{[i]}(r)r|\mathscr{U}|$
$\mathscr{R}_0 = \mathscr{H}_0+\mathscr{P}_0$
for $k=1\ldots\frac{t_f}{r}-1$ do
    $\mathscr{R}_k = \overline{\mathscr{M}}(r)\mathscr{R}_{k-1}+\mathscr{P}_0$
end for
$\mathscr{R}([0,t_f]) = \bigcup_{k=1}^{t_f/r}\mathscr{R}_{k-1}$
```

4.4 Overapproximating the State Transition Matrix

The computation of the set of possible state transition matrices $\{e^{\mathbf{A}t}|\mathbf{A}\in\mathscr{A}\}$ using matrix zonotopes and interval matrices as representation of the matrix set $\mathscr{A}$ are discussed next. Matrix zonotopes are more general than interval matrices, while the presented computations are more efficient using interval matrices. The presented techniques still work when the set of matrices $\mathscr{A}$ contains matrices for which the linear system is unstable. This is useful when considering hybrid systems with switched linear dynamics, where some linear systems are unstable, while the overall dynamics is stable.

4.4.1 Matrix Zonotopes

A matrix zonotope is defined as

$$\mathscr{A}_{[z]} = \left\{ \mathbf{G}^{(0)} + \sum_{i=1}^{\kappa} p_i \mathbf{G}^{(i)} \middle| p_i \in [-1,1], \mathbf{G}^{(i)} \in \mathbb{R}^{n\times n} \right\} \tag{4.8}$$

and is written in short form as $\mathscr{A}_{[z]} = (\mathbf{G}^{(0)}, \mathbf{G}^{(1)}, \ldots, \mathbf{G}^{(\kappa)})$, where the first matrix is referred to as the *matrix center* and the other matrices as *matrix generators*. The order of a matrix zonotope is defined as $\rho = \kappa/n$. When exchanging the matrix generators by vector generators $g^{(i)} \in \mathbb{R}^n$, one obtains a zonotope (see e.g. [14]). Matrix zonotopes can also be represented as the convex hull of its so-called matrix vertices $\mathbf{V}^{(i)}$:

$$\mathscr{A}_{[z]} = \left\{ \sum_{i=1}^{r} \alpha_i \mathbf{V}^{(i)} \middle| \mathbf{V}^{(i)} \in \mathbb{R}^{n\times n}, \alpha_i \in \mathbb{R}, \alpha_i \geq 0, \sum_i \alpha_i = 1 \right\}. \tag{4.9}$$

In order to obtain the Taylor series terms in (4.2), one has to compute the power of matrix zonotopes. This is done iteratively by $\mathscr{A}^l_{[z]} = \mathscr{A}_{[z]}\mathscr{B}_{[z]}$, where $\mathscr{B}_{[z]} = \mathscr{A}^{l-1}_{[z]}$. Thus, it suffices to show the multiplication of two matrix zonotopes $\mathscr{A}_{[z]} = (\mathbf{G}^{(0)}, \ldots, \mathbf{G}^{(\kappa_A)})$ and $\mathscr{B}_{[z]} = (\mathbf{H}^{(0)}, \ldots, \mathbf{H}^{(\kappa_B)})$:

$$\begin{aligned} \mathscr{A}_{[z]}\mathscr{B}_{[z]} &= \left[\!\!\left[\left(\mathbf{G}^{(0)} + \sum_{i=1}^{\kappa_A} p_i \mathbf{G}^{(i)}\right)\left(\mathbf{H}^{(0)} + \sum_{j=1}^{\kappa_B} q_j \mathbf{H}^{(j)}\right) \right]\!\!\right]_{p_i, q_j \in [-1,1]} \\ &= \mathbf{G}^{(0)}\mathbf{H}^{(0)} + \sum_{\substack{i=0 \\ (i,j)\neq(0,0)}}^{\kappa_A} \sum_{j=0}^{\kappa_B} \underbrace{[\![p_i q_j]\!]_{p_i,q_j\in[-1,1]}}_{\subseteq[-1,1]} \mathbf{G}^{(i)}\mathbf{H}^{(j)}, \end{aligned} \tag{4.10}$$

so that $\mathscr{A}_{[z]}\mathscr{B}_{[z]} \subseteq (\mathbf{G}^{(0)}\mathbf{H}^{(0)}, \mathbf{G}^{(0)}\mathbf{H}^{(1)}, \ldots, \mathbf{G}^{(\kappa_A)}\mathbf{H}^{(\kappa_B)})$. The Taylor terms up to second order are evaluated exactly:

Proposition 4.3 (Dependent Matrix Zonotope Evaluation). *The set* $[\![\mathbf{I} + \mathbf{A}t + 1/2(\mathbf{A}t)^2]\!]_{\mathbf{A}\in\mathscr{A}_{[z]}}$, *where* $\mathscr{A}_{[z]} = (\mathbf{G}^{(0)}, \mathbf{G}^{(1)}, \ldots, \mathbf{G}^{(\kappa_A)})$ *is enclosed by the smallest possible zonotope* $\mathscr{W}_{[z]}(t) = (\mathbf{L}^{(0)}(t), \mathbf{L}^{(1)}(t), \ldots, \mathbf{L}^{(\kappa_W)}(t))$, *where*

$$\begin{array}{lll} & \mathbf{L}^{(0)}(t) & = \mathbf{I} + \mathbf{G}^{(0)} t + \left({\mathbf{G}^{(0)}}^2 + \sum_{i=1}^{\kappa_A} 0.5 {\mathbf{G}^{(i)}}^2 \right) t^2, \\ j = 1 \dots \kappa_A : & \mathbf{L}^{(j)}(t) & = \mathbf{G}^{(j)} t + (\mathbf{G}^{(0)} \mathbf{G}^{(j)} + \mathbf{G}^{(j)} \mathbf{G}^{(0)}) t^2, \\ j = 1 \dots \kappa_A : & \mathbf{L}^{(\kappa_A + j)}(t) & = 0.5 {\mathbf{G}^{(j)}}^2 t^2, \\ l = \sum_{j=1}^{\kappa_A - 1} \sum_{k=j+1}^{\kappa_A} 1 : & \mathbf{L}^{(2\kappa_A + l)}(t) & = (\mathbf{G}^{(j)} \mathbf{G}^{(k)} + \mathbf{G}^{(k)} \mathbf{G}^{(j)}) t^2. \end{array}$$

Proof. The result of the multiplication $(\mathbf{G}^{(0)} + \sum_{i=1}^{\kappa_A} p_i \mathbf{G}^{(i)})(\mathbf{G}^{(0)} + \sum_{i=1}^{\kappa_A} p_i \mathbf{G}^{(i)})$ can be rearranged to

$$\begin{aligned} & {\mathbf{G}^{(0)}}^2 + \sum_{j=1}^{\kappa_A} p_j (\mathbf{G}^{(0)} \mathbf{G}^{(j)} + \mathbf{G}^{(j)} \mathbf{G}^{(0)}) + \sum_{j=1}^{\kappa_A} p_j^2 {\mathbf{G}^{(j)}}^2 \\ & + \sum_{j=1}^{\kappa_A - 1} \sum_{k=j+1}^{\kappa_A} p_j p_k (\mathbf{G}^{(j)} \mathbf{G}^{(k)} + \mathbf{G}^{(k)} \mathbf{G}^{(j)}), \end{aligned}$$

where $p_j, p_k \in [-1, 1]$ and $p_j^2 \in [0, 1]$. Since the interval $[0, 1]$ deviates from $[-1, 1]$ used as factors for matrix generators, it is split into $0.5 + [-1, 1]0.5$; this makes it possible to add the matrices $0.5 {\mathbf{G}^{(j)}}^2$ to the constant solution ${\mathbf{G}^{(0)}}^2$, and use the same matrix values as generator matrices. Applying this result to $[\![\mathbf{I} + \mathbf{A}t + 1/2(\mathbf{A}t)^2]\!]_{\mathbf{A} \in \mathscr{A}_{[z]}}$ results in the above proposition. □

4.4.2 Interval Matrices

An interval matrix is a special case of a matrix zonotope and specifies for each matrix element the interval of possible values:

$$\mathscr{A}_{[i]} = [\underline{\mathbf{A}}, \overline{\mathbf{A}}], \quad \forall i, j : \underline{a}_{ij} \leq \overline{a}_{ij}, \quad \underline{\mathbf{A}}, \overline{\mathbf{A}} \in \mathbb{R}^{n \times n}.$$

The matrix $\underline{\mathbf{A}}$ is referred to as the *lower bound* and $\overline{\mathbf{A}}$ as the *upper bound* of $\mathscr{A}_{[i]}$.

When computing with intervals, one generally uses interval arithmetic. In this work, only the addition and multiplication rule are required:

$$\begin{aligned} [a] + [b] &= [\underline{a} + \underline{b}, \overline{a} + \overline{b}], \\ [a] \cdot [b] &= [\min(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b}), \max(\underline{a}\underline{b}, \underline{a}\overline{b}, \overline{a}\underline{b}, \overline{a}\overline{b})]. \end{aligned} \tag{4.11}$$

For the computation of the Taylor terms $\frac{1}{i!}(\mathscr{A}_{[i]} t)^i$, one has to compute the power of interval matrices. This is done iteratively as for matrix zonotopes by ${\mathscr{A}_{[i]}}^l = \mathscr{A}_{[i]} \mathscr{B}_{[i]}$, where $\mathscr{B}_{[i]} = {\mathscr{A}_{[i]}}^{l-1}$. Using interval arithmetic, $\mathscr{C}_{[i]} = \mathscr{A}_{[i]} \mathscr{B}_{[i]}$ is computed elementwise by the single-use expression $[c_{ij}] = \sum_{k=1}^{n} [a_{ik}][b_{kj}]$, i.e. each matrix value occurs only once for each computation of $[c_{ij}]$. In interval arithmetic, single use expressions are always exact, e.g. $[a]([b] + 1) = [\![a(b + 1)]\!]_{a \in [a], b \in [b]}$. However, in this case, $\mathscr{B}_{[i]}$ is a function of $\mathscr{A}_{[i]}$ such that $\mathscr{C}_{[i]} \supseteq \mathscr{A}_{[i]} \mathscr{B}_{[i]}$.

In [36] it has been shown that the square of an interval matrix can be rewritten as a single-use expression, making the computation exact using interval arithmetic, i.e. the tightest possible interval matrix is computed. It has been further proven in [36] that it is NP-hard to compute the tightest enclosing interval matrix of the cube of an interval matrix ($\mathscr{A}_{[i]}{}^3$). The idea of computing the square of an interval matrix is extended in order to write as many computations of $[\![\mathbf{I}+\mathbf{A}t+1/2(\mathbf{A}t)^2]\!]_{\mathbf{A}\in\mathscr{A}_{[i]}}$ as a single-use expression, while the other expressions are evaluated by computing the global maxima.

Proposition 4.4 (Dependent Interval Matrix Evaluation). *The set* $[\![\mathbf{I}+\mathbf{A}t+1/2(\mathbf{A}t)^2]\!]_{\mathbf{A}\in\mathscr{A}_{[i]}}$ *can be tightly enclosed by another interval matrix* $\mathscr{W}_{[i]}(t) = [\underline{\mathbf{W}}(t), \overline{\mathbf{W}}(t)]$, *where*

$$\begin{aligned}
\forall i \neq j : [w_{ij}] =& [a_{ij}](t + \frac{1}{2}([a_{ii}] + [a_{jj}])t^2) + \frac{1}{2}\sum_{k:k\neq i, k\neq j} [a_{ik}][a_{kj}]t^2 \\
\forall i : [w_{ii}] =& \Big[\kappa([a_{ii}],t), \max\big(\underline{a}_{ii}t + \frac{1}{2}\underline{a}_{ii}^2t^2, \overline{a}_{ii}t + \frac{1}{2}\overline{a}_{ii}^2t^2\big)\Big] + \frac{1}{2}\sum_{k:k\neq i}[a_{ik}][a_{ki}]t^2 \\
\kappa([a_{ii}],t) =& \begin{cases} \min\big(\{\underline{a}_{ii}t + \frac{1}{2}\underline{a}_{ii}^2t^2, \overline{a}_{ii}t + \frac{1}{2}\overline{a}_{ii}^2t^2\}\big), \text{ for} -\frac{1}{t} \notin [a_{ii}] \\ -\frac{1}{2}, \text{ for} -\frac{1}{t} \in [a_{ii}] \end{cases}
\end{aligned}$$

Proof. The non-diagonal elements $[w_{ij}]$ can be formulated as a single-use expression (SUE), resulting in an exact evaluation using interval arithmetic. The computation of the diagonal elements $[w_{ii}]$ cannot entirely be reformulated to a SUE. However, one can split $[w_{ii}]$ into a part with and without a single variable occurrence:

$$[w_{ii}] = \underbrace{[a_{ii}]t + \frac{1}{2}[a_{ii}]^2t^2}_{\text{non-SUE}} + \underbrace{\frac{1}{2}\sum_{k:k\neq i}[a_{ik}][a_{ki}]t^2}_{\text{SUE}}.$$

It remains to obtain the exact interval of $\gamma(a) := at + \frac{1}{2}a^2t^2$ by computing the minimum and maximum. The function $\gamma(a)$ has only one minimum at $a = -1/t$ and is monotone elsewhere, so that the maximum is to be found at the borders: $\gamma_{max} = \max(\underline{a}_{ii}t + \frac{1}{2}\underline{a}_{ii}^2t^2, \overline{a}_{ii}t + \frac{1}{2}\overline{a}_{ii}^2t^2)$. Where the global minimum ($a_{min} = -1/t$) is an element of $[a_{ii}]$, one obtains: $\gamma_{min} = -1/2$. In the other case, the minimum is to be found at the border: $\gamma_{min} = \min(\underline{a}_{ii}t + \frac{1}{2}\underline{a}_{ii}^2t^2, \overline{a}_{ii}t + \frac{1}{2}\overline{a}_{ii}^2t^2)$. □

Besides computing with the lower and upper bound of intervals, one can also compute with the center and the radius of the interval. The advantage of the latter technique is that it is more efficient and easier to parallelize; see [31]. The result is more conservative, but the interval of a standard operation (addition, difference, multiplication, and division) is bounded by a factor 1.5 in radius[1] compared to the computation with lower and upper bounds.

[1] The radius of a set X is defined as $0.5\max_{x_1\in X, x_2\in X}|x_1 - x_2|$ in [31].

4.4.3 Norm Bounds

In order to quickly estimate the size of the set of state transition matrices, it is often helpful to compute with norms instead of applying the introduced computational techniques using matrix zonotopes or interval matrices.

Theorem 4.2 (Norm Bound). *In order to obtain a tight norm bound, the matrix set $\mathscr{A}$ is overapproximated by an interval matrix $\mathscr{A}_{[i]}$ which is split into a nominal and a symmetric part: $\mathscr{A}_{[i]} = \mathbf{A}_{[n]} + [-\mathbf{S}, \mathbf{S}]$. The norm of the distance of the set of state transition matrices to the exponential matrix of the nominal matrix is computed for $\| |\mathbf{A}_{[n]}| + \mathbf{S} \| < \frac{2}{t}$ as*

$$\begin{aligned} &\| \llbracket e^{\mathbf{A}t} \rrbracket_{\mathbf{A}\in\mathscr{A}} - e^{\mathbf{A}_{[n]}t} \| \\ &\leq \frac{\|\mathbf{A}_{[n]}\| \, \|\mathbf{S}\| \frac{t^2}{2}}{\|\mathbf{A}_{[n]}\| \cdot \| |\mathbf{A}_{[n]}| + \mathbf{S} \| \frac{t^2}{4} - (\|\mathbf{A}_{[n]}\| + \| |\mathbf{A}_{[n]}| + \mathbf{S} \|) \frac{t}{2} + 1} + \frac{\|\mathbf{S}\| t}{1 - \| |\mathbf{A}_{[n]}| + \mathbf{S} \| \frac{t}{2}}. \end{aligned}$$

The proof is shown in the Appendix.

4.4.4 Discussion

For small times $t < 2/(\| |\mathbf{A}_{[n]}| + \mathbf{S} \|)$ (see the Appendix), which are typically used for reachability analysis, the terms $\frac{1}{i!}(\mathscr{A}t)^i$ contribute less to the overall solution $\llbracket e^{\mathbf{A}t} \rrbracket_{\mathbf{A}\in\mathscr{A}}$ for increasing i values. Thus, one should use sophisticated computations for the first terms and switch to coarser and more efficient computations for higher order terms. For this reason, computations with matrix zonotopes are only conducted up to second order in this work. Another reason is that the number of generators for the l^{th} power is $(\kappa + 1)^l - 1$, while the representation size does not grow for interval matrices. In order to keep the overapproximation of interval computations low, higher powers are based on the exact result of the square; see [36]. Besides matrix zonotopes, one can also represent uncertainties by the more general matrix polytopes [27]. However, due to the computational complexity of matrix polytopes, it is advisable to overapproximate them by matrix zonotopes (see [27]) and compute with the methods presented herein.

4.4.5 Numerical Evaluation of the Set of State Transition Matrices

The methods presented for computing the set of state transition matrices are illustrated for a five-dimensional example and evaluated for randomly generated examples.

4.4.5.1 Five-Dimensional Example

The computation of the set of state transition matrices is demonstrated for the matrix zonotope

$$\mathscr{A}_{[z]} = (\mathbf{G}^{(0)}, \mathbf{G}^{(1)}), \quad \mathbf{G}^{(0)} = \begin{bmatrix} -1 & -4 & 0 & 0 & 0 \\ 4 & -1 & 0 & 0 & 0 \\ 0 & 0 & -3 & 1 & 0 \\ 0 & 0 & -1 & -3 & 0 \\ 0 & 0 & 0 & 0 & -2 \end{bmatrix}, \mathbf{G}^{(1)} = \begin{bmatrix} 0.1 & 0.1 & 0 & 0 & 0 \\ 0.1 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0.1 & 0 \\ 0 & 0 & 0.1 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}. \tag{4.12}$$

and the corresponding interval matrix that tightly encloses the above matrix zonotope:

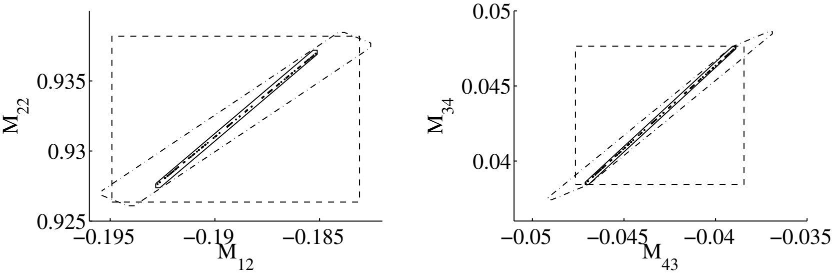
$$\mathscr{A}_{[i]} = \begin{bmatrix} [-1.1,-0.9] & [-4.1,-3.9] & 0 & 0 & 0 \\ [3.9,4.1] & [-1.1,-0.9] & 0 & 0 & 0 \\ 0 & 0 & [-3.1,-2.9] & [0.9,1.1] & 0 \\ 0 & 0 & [-1.1,-0.9] & [-3.1,-2.9] & 0 \\ 0 & 0 & 0 & 0 & [-2.1,-1.9] \end{bmatrix}. \tag{4.13}$$

The resulting sets $\overline{\mathscr{M}}(t)$ are computed for $t = 0.05$ and the maximum order $\eta = 6$ of the Taylor expansion. For the matrix zonotope $\mathscr{A}_{[z]}$, the set of state transition matrices is plotted for selected projections in Fig. 4.3. Particular matrix exponential values generated from matrix samples $\check{\mathbf{A}}_i \in \mathscr{A}_{[z]}$ are also plotted. These matrices are the vertex matrices of $\mathscr{A}_{[z]}$ and 100 randomly chosen matrices. One can observe that the matrix zonotope computation is much more accurate and captures very well the result of the samples, while the interval matrix computation returns a much larger set. The independent evaluation of each Taylor term using matrix zonotopes, i.e. (4.10) is applied for the first two Taylor terms instead of Prop. 4.3, also returns a much larger set compared to the dependent evaluation of the terms up to second order.

The results for the interval matrix $\mathscr{A}_{[i]}$ are shown in Fig. 4.4. Obviously, the computation with matrix zonotopes results only in marginal improvements when the uncertain matrix is an interval matrix, while it is a more significant improvement over the independent evaluation, i.e. pure interval arithmetic is applied for the first two Taylor terms instead of Prop. 4.4. For interval matrices, the result is tight for both, the interval matrix and the matrix zonotope computation.

4.4.5.2 Random Matrix Set Generation

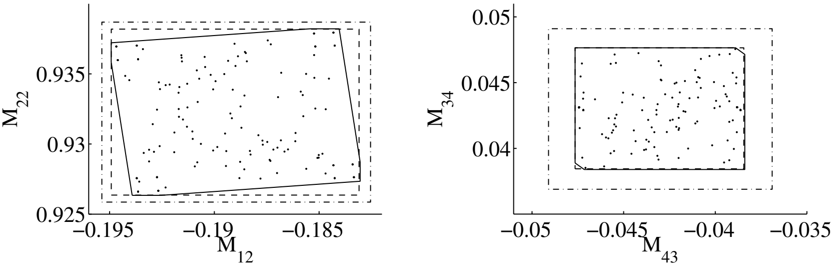
For a more thorough evaluation, random matrix sets are computed using a number of characterizing parameters. A random matrix whose elements are uniformly distributed is denoted by $\mathbf{A}^{\mathrm{rand}} = \mathtt{rand}(\overline{a}, \mu)$ so that $\forall i, j : -\overline{a} \leq a_{ij}^{\mathrm{rand}} \leq \overline{a}$. The variable μ determines the ratio of the number of non-zero elements to all elements of a



(a) Projection onto $\overline{\mathscr{M}}_{12}, \overline{\mathscr{M}}_{22}$.

(b) Projection onto $\overline{\mathscr{M}}_{43}, \overline{\mathscr{M}}_{34}$.

Fig. 4.3: Computations of $\overline{\mathscr{M}(t)}$ for the set $\mathscr{A}_{[z]}$ as specified in (4.12); $t = 0.05$, $\eta = 6$. Solid line: matrix zonotope computation; dashed line: interval matrix computation; dash-dotted line: independent matrix zonotope computation, i.e. independent evaluation of each Taylor term



(a) Projection onto $\overline{\mathscr{M}}_{12}, \overline{\mathscr{M}}_{22}$.

(b) Projection onto $\overline{\mathscr{M}}_{43}, \overline{\mathscr{M}}_{34}$.

Fig. 4.4: Computations of $\overline{\mathscr{M}(t)}$ for the set $\mathscr{A}_{[i]}$ as specified in (4.13); $t = 0.05$, $\eta = 6$. Solid line: matrix zonotope computation; dashed line: interval matrix computation; dash-dotted line: independent matrix zonotope computation, i.e. independent evaluation of each Taylor term

matrix, i.e. the number of non-zero values is $\texttt{ceil}(\mu n^2)$ and `ceil` returns the next higher natural number.

The matrix center and matrix generators are randomly generated as $\mathbf{G}^{(0)} = \texttt{rand}(\sigma, 1)$ and $\mathbf{G}^{(i)} = \texttt{rand}(\frac{1}{\kappa}, \mu)$, where σ is referred to as center-uncertainty ratio, κ is the number of generators, and μ is the non-zero ratio. Note that the non-zero elements have the same row and column indices for all generator matrices so that the corresponding interval matrix uncertainties are non-zero at the same positions. The interval enclosure of matrix zonotopes is equivalent to generating interval matrices $\mathbf{G}^{(0)} + [-\mathbf{S}, \mathbf{S}]$, where $\mathbf{S} = \texttt{rand}(1, \mu)$.

There are no further constraints on the generation of random matrix sets, such that the sets might contain stable and/or unstable matrices.

4.4.5.3 Norm Evaluation

As a first test, the norm $\|\mathscr{M}(t) - \mathbf{M}_{[n]}(t)\|_\infty$ with $\mathbf{M}_{[n]}(t) = e^{\mathbf{A}_{[n]}t}$ as defined in (4.3) is over- and underapproximated. The underapproximation is obtained as a union of sampled matrices: $\underline{\mathscr{M}}(t) = \bigcup_{i=1}^{\varpi} e^{\check{\mathbf{A}}^{(i)}t}$, where $\check{\mathbf{A}}^{(i)}$ are vertex matrices and 10^3 randomly generated matrices.

The overapproximation is obtained as presented above and the inf-norm when computing with interval matrices can easily be computed as $\|\mathscr{A}_{[i]}\|_\infty = \|A^*\|_\infty$, where $a^*_{ij} = \max(|\underline{a}_{ij}|, |\overline{a}_{ij}|)$. Note that computing the 2-norm of an interval matrix is exponential in the system dimension [37]. When the set of uncertain matrices is from the class of matrix zonotopes, the maximum norm is to be found equal to one of the vertex matrices $\mathbf{V}^{(i)}$ since $\|\sum_{i=1}^{r_A} \alpha_i \mathbf{V}^{(i)}\| \leq \sum_{i=1}^{r_A} \alpha_i \|\mathbf{V}^{(i)}\|$, $\alpha_i \geq 0$ (see (4.9)). However, the number of vertices is too high, even in small dimensions, such that only interval matrices can be evaluated. This is obvious since already the number of vertex matrices required to represent the remainder $\mathscr{E}_{[i]}$ is 2^{n^2} when each element of $\mathscr{E}_{[i]}$ is uncertain within an interval.

The ratio of both norms is defined as

$$\theta = \frac{\|\overline{\mathscr{M}}(t) - \mathbf{M}_{[n]}(t)\|_\infty}{\|\underline{\mathscr{M}}(t) - \mathbf{M}_{[n]}(t)\|_\infty}$$

and its evaluation is performed using randomly generated interval matrices with parameters specified in Table 4.1. After introducing $t_{\max} = 2/\|\mathscr{A}_{[i]}\|_\infty$, one can define the time-ratio $\omega := t/t_{\max}$ so that for $\omega \in [0,1]$ the convergence of the norm bound is guaranteed (see the Appendix). By varying one of the parameters while fixing the others, and by choosing the maximum Taylor order to $\eta = 10$, the plots in Fig. 4.5 are obtained. It can be seen that the only dominant parameter is the time ratio ω, while all other variations return norm ratios of around 1.2 which is mainly caused by choosing $\omega = 0.2$.

Table 4.1: Error norm evaluation: Random matrix set generation parameters

dimension n	center-delta ratio σ	time ratio ω	non-zero ratio μ
20	3	0.2	0.3

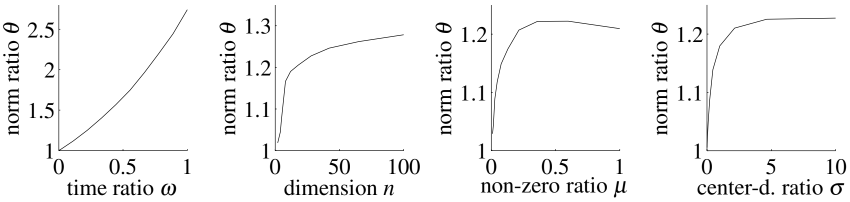


Fig. 4.5: Norm evaluation: Norm ratios θ for variations of parameters while fixing the other parameters given in Table 4.1

4.4.5.4 Volume Evaluation

Since the performance of the matrix zonotope computations could not be evaluated in the previous norm test, we now evaluate how big the volume of the set of state transition matrices $\overline{\mathscr{M}}(t)$ is when it is computed by matrix zonotopes or interval matrices. The volume of $\overline{\mathscr{M}}(t)$ is computed by transforming it to a set in the vector space, so that interval matrices become multidimensional intervals and matrix zonotopes become zonotopes. The transformation is established by stacking the column vectors of a matrix $\mathbf{Y} \in \mathbb{R}^{n\times n}$ into a vector $\mathbf{y} \in \mathbb{R}^{n^2}$.

The volume computation of multidimensional intervals is simply the product of the interval lengths in each dimension. The volume computation of a zonotope is more elaborate and $\sharp P$-hard; see [38]. For this reason, zonotopes are overapproximated by parallelotopes according to [5] for which the volume computation is much easier, meaning that the exact volume ratio is better for matrix zonotopes than shown in Fig. 4.6. In order to ensure that the volume is always greater than 0, the non-zero ratio μ is chosen to 1. Due to the computational load, the dimension is chosen as $n = 6$ in contrast to Table 4.1. For a comparison of the results, the average ratio for each dimension is computed: $\upsilon = (V_1/V_2)^{1/n^2}$, where V_1 is the volume of the zonotope computation, V_2 the volume of the interval computation, and the dimension due to the vector space transformation is n^2. It can be seen that especially for problems in higher dimension, matrix zonotopes perform better than interval matrices.

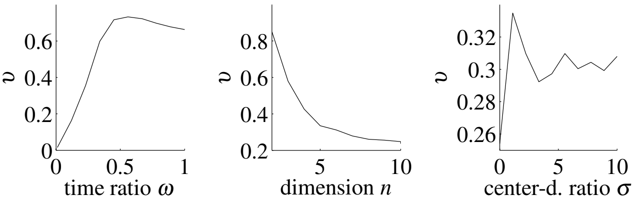


Fig. 4.6: Volume evaluation: Normalized volume ratios υ for variations of parameters while fixing the other parameters.

4.5 Computation of Reachable Sets

As mentioned in the introduction, there exists a large number of possible representations for reachable sets. It has been shown that zonotopes and support functions outperform other representations when computing the reachable set of linear time invariant systems [16, 39]. However, for linear systems with uncertain parameters, no efficient method has yet been proposed using support functions. Thus, zonotopes are used for the numerical examples, which are specified as in (4.8), except that the matrix generators are replaced by vector generators. The order of a zonotope is also defined as $\rho = \kappa/n$, where κ is the number of generators and n is the system dimension.

In order to execute Alg. 1, it remains to specify how to multiply an interval matrix or a matrix zonotope with a zonotope, and how to add zonotopes. Due to space limitations, the derivation of these operations is left to [27]. It is noted that the multiplication and addition operation can be implemented efficiently which is reflected in the numerical examples presented below.

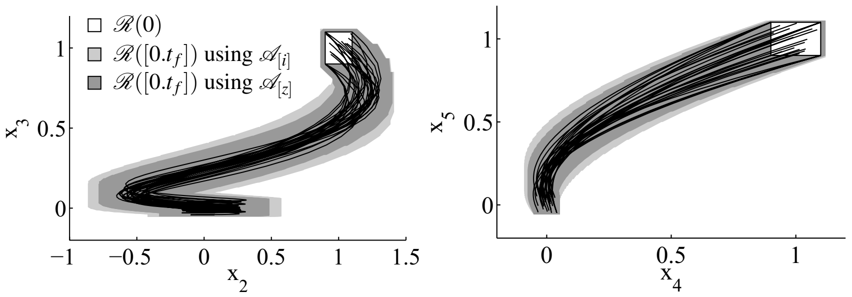
4.5.1 Five-Dimensional Example

As a first example, the reachable set of the linear system $\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{u}(t)$ is computed, where $\mathbf{A} \in \mathscr{A}_{[z]}$ as specified in (4.12). Alternatively, the reachable set is computed with interval matrices so that $A \in \mathscr{A}_{[i]}$ as specified in (4.13) to compare the accuracy with the more complex matrix zonotope computations. The set of inputs is bounded by the interval $[-0.1, 0.1]$ for each dimension. The maximum order of Taylor terms is chosen to $\eta = 4$, the maximum zonotope order is chosen as $\rho = 20$, the time increment is $r = 0.05$ and the time horizon is $t_f = 5$.

The scalability of the algorithm is shown by computing reachable sets for several randomly generated linear systems using the same parameters as for the five-dimensional system. There are no further constraints on the generation of random matrix sets, such that the sets might contain stable and/or unstable matrices. Computation times for system matrices bounded by interval matrices and matrix zonotopes are shown in Table 4.2. The computations have been performed in MATLAB on an Intel i7 Processor with 1.6 GHz and 6 GB memory.

4.5.2 Transmission Line

The second example is a transmission line which is modeled as an R-L-C circuit, see Fig. 4.8. Those models are used in, e.g., timing verification of integrated circuit design [40]. Possible verification tasks are to guarantee a minimum time to reach a certain output voltage or to guarantee that a maximum output voltage is not overshot. Similar examples have been studied in [16, 41], where wrapping-free algorithms



(a) Projection onto x_2, x_3.

(b) Projection onto x_4, x_5.

Fig. 4.7: Reachable set of the five-dimensional example. The light gray region shows the reachable set when computing with the interval matrix $\mathscr{A}_{[i]}$, while the dark gray region shows the result when computing with the original matrix zonotope $\mathscr{A}_{[z]}$. Black lines show exemplary trajectories and the white region is the initial set

Table 4.2: Computation times

Dimension n	5	10	20	50	100
Interval matrix					
CPU time in [s]	0.14	0.17	0.46	1.05	3.63
Matrix zonotope: Nr of generator matrices $\kappa = 1$					
CPU time in [s]	0.14	0.15	0.46	1.36	6.24
Matrix zonotope: Nr of generator matrices $\kappa = 2$					
CPU time in [s]	0.15	0.20	0.72	3.53	11.01
Matrix zonotope: Nr of generator matrices $\kappa = 4$					
CPU time in [s]	0.22	0.36	1.47	7.58	28.33

could be applied. This is not possible in this work since uncertain parameters are considered. The wrapping effect plays a dominant role in this example since the system is poorly damped, where the smallest damping ratio of all poles is 0.016. Thus, even a small wrapping effect can cause unstable reachable set computations. This effect could be decreased by applying subdivision strategies for the uncertain parameters, which would increase the computation time, however.

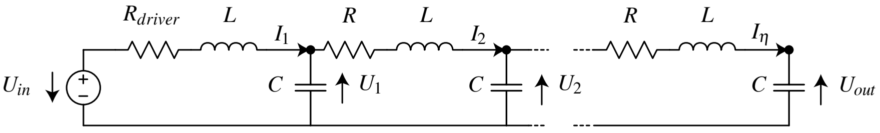


Fig. 4.8: Transmission line modeled as an R-L-C circuit

After denoting the voltage and the current at the l^{th} node by U_l and I_l, respectively, and all resistances, inductances, and capacitances by R, L, C, the differential equations are

first node ($l = 1$)	other nodes	last node ($l = \eta$)
$\dot{U}_1 = \frac{1}{C}(I_2 - I_1)$	$\dot{U}_l = \frac{1}{C}(I_{l+1} - I_l)$	$\dot{U}_\eta = -\frac{1}{C}I_\eta$
$\dot{I}_1 = \frac{1}{L}(U_1 + U_{in}) - \frac{R_{driver}}{L}I_1$	$\dot{I}_l = \frac{1}{L}(U_l - U_{l-1}) - \frac{R}{L}I_l$	$\dot{I}_\eta = \frac{1}{L}(U_\eta - U_{\eta-1}) - \frac{R}{L}I_\eta$

(4.14)

with parameter ranges listed in Table 4.3. After introducing the state vector $\mathbf{x} = [U_1, \ldots, U_\eta, I_1, \ldots, I_\eta]^T$, the input $u = U_{in}$, and grouping the terms in (4.14), one can formulate the differential inclusion

$$\dot{\mathbf{x}} \in \underbrace{([p_1]\mathbf{Q}^{(1)} + [p_2]\mathbf{Q}^{(2)} + [p_3]\mathbf{Q}^{(3)} + [p_4]\mathbf{Q}^{(4)})}_{=\mathscr{A}_{[z]}}\mathbf{x} + \underbrace{[p_1]\mathbf{r}}_{=\mathscr{B}_{[z]}}u, \tag{4.15}$$

where $\mathbf{Q}^{(i)} \in \mathbb{R}^{n \times n}$, $\mathbf{r} \in \mathbb{R}^n$, and

$$[p_1] = \frac{1}{[L]}, \quad [p_2] = \frac{1}{[C]}, \quad [p_3] = \frac{[R_{driver}]}{[L]}, \quad [p_4] = \frac{[R]}{[L]}.$$

The formulation in (4.15) makes it possible to obtain the generators $\mathbf{G}^{(i)}$ of the matrix zonotope $\mathscr{A}_{[z]}$ as

$$\mathbf{G}^{(0)} = \sum_{i=1}^{4} \texttt{mid}\{[p_i]\}\mathbf{Q}^{(i)}, \text{ for } i = 1..4 : \mathbf{G}^{(i)} = \texttt{rad}\{[p_i]\}\mathbf{Q}^{(i)}$$

and analogously for $\mathscr{B}_{[z]}$, where $\texttt{mid}\{.\}$ returns the midpoint and $\texttt{rad}\{.\}$ the radius of an interval. The initial state of the system is determined by the steady state solution for input voltages $U_{in} = u \in [-0.2, 0.2]$ to which an uncertainty is added so that the initial currents are also uncertain: $R(0) = -\mathbf{A}^{-1}\mathbf{b}u + \square(0.001)$, where $\mathbf{A}$, $\mathbf{b}$ are chosen as the matrix centers of $\mathscr{A}_{[z]}$, $\mathscr{B}_{[z]}$, and $\square(0.001)$ is a box of edge length $2 \cdot 0.001$. At time $t = 0$, the input is changed to $u \in [0.99, 1.01]$ so that the step response of the output voltage $U_{out} = U_l$ can be verified. For the modeling of the transmission line, 20 nodes have been used such that the system has 40 state variables. The reachable set of U_{out} is presented in Fig. 4.9 when computing with matrix zonotopes (dark gray) or interval matrices (light gray). It can be observed that the matrix zonotope computations are much tighter due to the consideration of the dependency of the R, L, and C values of each node. Further projections of reachable sets in the phase space are shown in Fig. 4.10.

The step size of the example is $r = 0.002$, the time horizon is $t_f = 0.7$, Taylor terms are computed up to order $\eta = 6$, and the maximum zonotope order is $\rho = 400$, where the order reduction is performed as in [14]. The computation time was 388 s for the matrix zonotope computation and 37 s for the interval matrix computation in MATLAB (without using the parallel computing toolbox) on an Intel i7 Processor with 1.6 GHz and 6 GB memory. Interval computations have been performed using the Matlab toolbox IntLab [42].

Table 4.3: Transmission Line Parameters

resistance in [Ω]	driver resistance in [Ω]	inductance in [H]	capacitance in [F]
$R \in [0.99, 1.01]$	$R_{driver} \in [9.9, 10.1]$	$L = 1e-10$	$C \in 1e-13 \cdot [3.99, 4.01]$

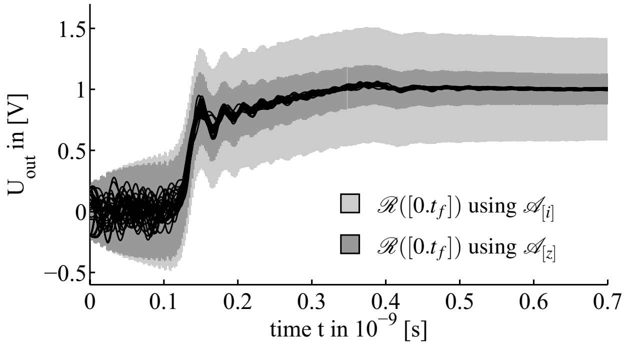
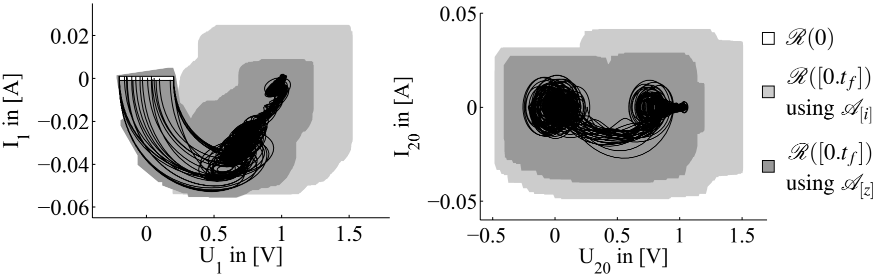


Fig. 4.9: Output voltage range of the transmission line over time. The light gray region shows the reachable set when computing with the interval matrix $\mathscr{A}_{[i]}$, while the dark gray region shows the result when computing with the original matrix zonotope $\mathscr{A}_{[z]}$. Black lines show exemplary trajectories

4.6 Conclusions

The computation of reachable sets for linear systems with uncertain time-invariant system matrices and time-varying inputs has been presented. The reachable set for points in time without any input is computed based on the set of state transition matrices, which is extended for time intervals and uncertain inputs. New methods for tightly overapproximating the set of state transition matrices by considering parameter dependencies have been developed for interval matrices and matrix zonotopes. These methods are numerically evaluated and supplemented by an accurate norm estimation. Due to the use of zonotopes for the reachable set representation, the computational complexity grows moderately with the number of state variables compared to other approaches, such as the computation with arbitrary polytopes. The usefulness of the presented methods is demonstrated for the verification of a



(a) Projection onto U_1, I_1.

(b) Projection onto U_{20}, I_{20}.

Fig. 4.10: Reachable set of the transmission line example. The light gray region shows the reachable set when computing with the interval matrix $\mathscr{A}_{[i]}$, while the dark gray region shows the result when computing with the original matrix zonotope $\mathscr{A}_{[z]}$. Black lines show exemplary trajectories and the white region is the initial set

transmission line. Although the overapproximation of reachable sets is small for the first time intervals, the wrapping effect might become a dominant source for overapproximation when the system is poorly damped.

As previously mentioned, it is assumed that the implementation of the presented methods returns exact numerical results, although computers have rounding errors due to a fixed number of significant digits. This can be fixed by performing all underlying numerical computations with interval arithmetics accounting for rounding errors.

Future work aims at reducing the wrapping effect by developing new order reduction techniques for zonotopes. This might be achieved by adopting techniques used for the reduction of the wrapping effect of multidimensional intervals, such as the QR-preconditioning algorithm [29]. Preconditioning the state equations such as using the classical diagonalization of system matrices, where $\mathscr{A}^* = S\mathscr{A}S^{-1}$ and S contains the eigenvectors of the nominal system matrix $\mathbf{A}_{[n]}$, has not been beneficial since the uncertainty of $\mathscr{A}^*$ is increased compared to $\mathscr{A}$ due to the necessary matrix set multiplications. However, if one could compute the range of eigenvalues and eigenvectors more efficiently and tighter as today [34], one could use these results to obtain an exactly diagonalized system matrix directly. It would then be possible to compute the set of state transition matrices for long time horizons (due to the separate evaluation for each dimension) such that wrapping-free implementation developed in [39] could be applied.

Acknowledgements The authors would like to thank Colas Le Guernic for providing the tighter computation of the state transition matrix remainder and the fruitful discussions. This research was supported in part by U.S. National Science Foundation grant number CCF-0926181 and the U.S. Air Force Office of Scientific Research grant number FA9550-06-1-0312.

Appendix

Proof of Theorem 4.2

The l^{th} power of an interval matrix can be represented by a real valued matrix $\mathbf{C}_{[n]}(l)$ and a symmetric interval matrix $[-\mathbf{D}(l),\mathbf{D}(l)]$:

$$\mathscr{A}_{[i]}{}^{l} = (\mathbf{A}_{[n]} + [-\mathbf{S},\mathbf{S}])^{l} = \mathbf{C}_{[n]}(l) + [-\mathbf{D}(l),\mathbf{D}(l)].$$

Using the nominal or center value $\mathbf{A}_{[n]}$ and the symmetric interval $[-\mathbf{S},\mathbf{S}]$, the values of $\mathbf{C}_{[n]}(l)$ and $\mathbf{D}(l)$ can be obtained iteratively (see [31]):

$$\begin{aligned} &\mathbf{C}_{[n]}(i+1) = \mathbf{A}_{[n]}\mathbf{C}_{[n]}(i), \\ &\quad \mathbf{D}(i+1) \leq |\mathbf{A}_{[n]}|\mathbf{D}(i) + \mathbf{S}|\mathbf{C}_{[n]}(i)| + \mathbf{S}\mathbf{D}(i) = (|\mathbf{A}_{[n]}| + \mathbf{S})\mathbf{D}(i) + \mathbf{S}|\mathbf{C}_{[n]}(i)|, \end{aligned} \tag{4.16}$$

where $\mathbf{C}_{[n]}(1) = \mathbf{A}_{[n]}$, $\mathbf{D}(1) = \mathbf{S}$. Using this notation, the difference between the nominal exponential matrix and the overapproximated set of exponential matrices is

$$[\![e^{\mathbf{A}t}]\!]_{\mathbf{A}\in\mathscr{A}} - e^{\mathbf{A}_{[n]}t} \subseteq \sum_{i=1}^{\infty}[-\mathbf{D}(i),\mathbf{D}(i)]\frac{t^i}{i!}. \tag{4.17}$$

We are ultimately interested in $\mathbf{S}^{\Sigma}(i) := \sum_{l=1}^{i}\mathbf{D}(l)\frac{t^l}{l!}$ (see (4.17)). A matrix computation can be found for $\mathbf{S}^{\Sigma}(i)$ based on (4.16) when overapproximating the absolute value of $\mathbf{C}_{[n]}(i)$ by $|\mathbf{C}_{[n]}(i+1)| = |\mathbf{A}_{[n]}||\mathbf{C}_{[n]}(i)|$:

$$\begin{bmatrix} |\mathbf{C}_{[n]}(i+1)|t^{i+1} \\ \mathbf{D}(i+1)t^{i+1} \\ \mathbf{S}^{\Sigma}(i+1) \end{bmatrix} = \underbrace{\begin{bmatrix} |\mathbf{A}_{[n]}|t & 0 & 0 \\ \mathbf{S}t & (|\mathbf{A}_{[n]}|+\mathbf{S})t & 0 \\ 0 & \mathbf{I}\frac{1}{i!} & \mathbf{I} \end{bmatrix}}_{=\tilde{\mathbf{G}}(i)} \begin{bmatrix} |\mathbf{C}_{[n]}(i)|t^{i} \\ \mathbf{D}(i)t^{i} \\ \mathbf{S}^{\Sigma}(i) \end{bmatrix}.$$

In order to derive some properties from the linear update scheme, the matrix $\tilde{\mathbf{G}}(i)$ depending on i is replaced by a constant matrix $\mathbf{G}$ such that the result is overapproximated. This is done by overapproximating the sum $\mathbf{S}^{\Sigma}(i) = \sum_{l=1}^{i}\mathbf{D}(l)\frac{t^l}{l!}$ by $\mathbf{S}^{\Sigma^*}(i) = \sum_{l=1}^{i}\mathbf{D}(l)\frac{t^l}{2^{l-1}}$:

$$\begin{bmatrix} |\mathbf{C}_{[n]}(i+1)|t^{i+1}/2^{i} \\ \mathbf{D}(i+1)t^{i+1}/2^{i} \\ \mathbf{S}^{\Sigma}(i+1) \end{bmatrix} = \underbrace{\begin{bmatrix} |\mathbf{A}_{[n]}|\frac{t}{2} & 0 & 0 \\ \mathbf{S}\frac{t}{2} & (|\mathbf{A}_{[n]}|+\mathbf{S})\frac{t}{2} & 0 \\ 0 & \mathbf{I} & \mathbf{I} \end{bmatrix}}_{=\mathbf{G}} \begin{bmatrix} |\mathbf{C}_{[n]}(i)|t^{i}/2^{i-1} \\ \mathbf{D}(i)t^{i}/2^{i-1} \\ \mathbf{S}^{\Sigma}(i) \end{bmatrix}. \tag{4.18}$$

When using the definition of norms of matrix sets in (4.3), the following relationships hold: $\|\mathbf{A}\,\mathbf{B}\| \leq \|\mathbf{A}\|\,\|\mathbf{B}\|$, $\|\mathbf{A}+\mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$. The inequality for the multiplication only holds for sub-multiplicative norms, which is the case for all p-norms. This makes it possible to rewrite (4.18) to

$$\begin{bmatrix} \|\mathbf{C}_{[n]}(i+1)\| t^{i+1}/2^{i} \\ \|\mathbf{D}(i+1)\| t^{i+1}/2^{i} \\ \|\mathbf{S}^{\Sigma}(i+1)\| \end{bmatrix} = \underbrace{\begin{bmatrix} \|\mathbf{A}_{[n]}\|\frac{t}{2} & 0 & 0 \\ \|\mathbf{S}\|\frac{t}{2} & \||\mathbf{A}_{[n]}|+\mathbf{S}\|\frac{t}{2} & 0 \\ 0 & 1 & 1 \end{bmatrix}}_{=\mathbf{G}_{norm}} \begin{bmatrix} \|\mathbf{C}_{[n]}(i)\| t^{i}/2^{i-1} \\ \|\mathbf{D}(i)\| t^{i}/2^{i-1} \\ \|\mathbf{S}^{\Sigma}(i)\| \end{bmatrix}$$

Due to the block-triangular structure of $\mathbf{G}_{norm}$, the eigenvalues are $\|\mathbf{A}_{[n]}\|\frac{t}{2}$, $\||\mathbf{A}_{[n]}|+\mathbf{S}\|\frac{t}{2}$, and 1. If $\||\mathbf{A}_{[n]}|+\mathbf{S}\|\frac{t}{2} < 1$, it follows that the maximum eigenvalue is 1, which is assumed from now on. Another interesting property of $\mathbf{G}_{norm}$ is that it is non-negative, i.e. $g_{norm,i,j} \geq 0$ for all $i, j = 1 \ldots 3$. In addition, if there exists a common natural number m for all index pairs such that $(\mathbf{G}^{m}_{norm})_{ij} > 0$, the matrix is not only irreducible, but primitive, too.

For primitive matrices, one can apply the Perron-Frobenius theorem that allows one to compute $\lim_{k\to\infty}\mathbf{G}^{k}_{norm}$ based on the left and right eigenvectors of $\mathbf{G}_{norm}$. However, due to the block-triangular structure of $\mathbf{G}_{norm}$, it follows that it is a reducible matrix and thus not primitive. In [43] it is shown that under certain conditions (see [43, Assumption 2]), the results of the Perron-Frobenius theorem can be generalized to the reducible matrix at hand, where $\mathbf{y}$ is the right and $\mathbf{q}$ the left eigenvector corresponding to the greatest eigenvalue $\bar{\lambda} = 1$: $\lim_{k\to\infty}\mathbf{G}^{k}_{norm}/\bar{\lambda}^{k} = \lim_{k\to\infty}\mathbf{G}^{k}_{norm} = \mathbf{y}\mathbf{q}^{T}/(\mathbf{y}^{T}\mathbf{q})$. Using this result, and the fact that the right eigenvector is always $\mathbf{y} = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^{T}$ and $q_3 = 1$, the norm of the set of matrix exponentials can be overapproximated by

$$\|[\![e^{\mathbf{A}t}]\!]_{\mathbf{A}\in\mathscr{A}} - e^{\mathbf{A}_{[n]}t}\| \leq \underbrace{\begin{bmatrix}0 & 0 & 1\end{bmatrix}\frac{\mathbf{y}\mathbf{q}^{T}}{\mathbf{q}^{T}\mathbf{y}}}_{=\mathbf{q}^{T}} \begin{bmatrix} \|\mathbf{A}_{[n]}\|t \\ \|\mathbf{S}\|t \\ 0 \end{bmatrix}. \tag{4.19}$$

The remaining left eigenvectors are

$$q_1 = \frac{\|\mathbf{S}\|\frac{t}{2}}{\|\mathbf{A}_{[n]}\|\cdot\||\mathbf{A}_{[n]}|+\mathbf{S}\|\frac{t^2}{4} - (\|\mathbf{A}_{[n]}\| + \||\mathbf{A}_{[n]}|+\mathbf{S}\|)\frac{t}{2} + 1},$$
$$q_2 = \frac{1}{1 - \||\mathbf{A}_{[n]}|+\mathbf{S}\|\frac{t}{2}}.$$

Inserting this result into (4.19) yields the result of the theorem.

References

1. Glover, J.D., Schweppe, F.C.: Control of linear dynamic systems with set constrained disturbances. IEEE Transactions on Automatic Control **16**(5), 411–423 (1971)

2. Clarke, E., Fehnker, A., Han, Z., Krogh, B.H., Ouaknine, J., Stursberg, O., Theobald, M.: Abstraction and counterexample-guided refinement in model checking of hybrid systems. International Journal of Foundations of Computer Science **14**(4), 583–604 (2003)
3. Schlaepfer, F.M., Schweppe, F.C.: Continuous-time state estimation under disturbances bounded by convex sets. IEEE Transactions on Automatic Control **17**(2), 197–205 (1972)
4. Althoff, M., Stursberg, O., Buss, M.: Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In: Proc. of the 47th IEEE Conference on Decision and Control, pp. 4042–4048 (2008)
5. Althoff, M., Stursberg, O., Buss, M.: Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes. Nonlinear Analysis: Hybrid Systems **4**(2), 233–249 (2010)
6. Henzinger, T.: Verification of Digital and Hybrid Systems, *NATO ASI Series F: Computer and Systems Sciences*, vol. 170, chap. The theory of hybrid automata, pp. 265–292. Springer (2000)
7. Henzinger, T.A., Ho, P.H., Wong-Toi, H.: Algorithmic analysis of nonlinear hybrid systems. IEEE Transactions on Automatic Control **43**(4), 540–554 (1998)
8. Frehse, G.: PHAVer: Algorithmic verification of hybrid systems past HyTech. In: Hybrid Systems: Computation and Control, LNCS 3413, pp. 258–273. Springer (2005)
9. Lafferriere, G., Pappas, G.J., Yovine, S.: Symbolic reachability computation for families of linear vector fields. Symbolic Computation **32**, 231–253 (2001)
10. Chutinan, A., Krogh, B.H.: Computational techniques for hybrid system verification. IEEE Transactions on Automatic Control **48**(1), 64–75 (2003)
11. Kurzhanskiy, A.B., Varaiya, P.: Ellipsoidal techniques for reachability analysis of discrete-time linear systems. IEEE Transactions on Automatic Control **52**(1), 26–38 (2007)
12. Stursberg, O., Krogh, B.H.: Efficient representation and computation of reachable sets for hybrid systems. In: Hybrid Systems: Computation and Control, LNCS 2623, pp. 482–497. Springer (2003)
13. Kühn, W.: Rigorously computed orbits of dynamical systems without the wrapping effect. Computing **61**, 47–67 (1998)
14. Girard, A.: Reachability of uncertain linear systems using zonotopes. In: Hybrid Systems: Computation and Control, LNCS 3414, pp. 291–305. Springer (2005)
15. Tomlin, C., Mitchell, I., Bayen, A., Oishi, M.: Computational techniques for the verification and control of hybrid systems. Proceedings of the IEEE **91**(7), 986–1001 (2003)
16. Girard, A., Guernic, C.L.: Efficient reachability analysis for linear systems using support functions. In: Proc. of the 17th IFAC World Congress, pp. 8966–8971 (2008)
17. Henzinger, T.A., Horowitz, B., Majumdar, R., Wong-Toi, H.: Beyond HyTech: Hybrid systems analysis using interval numerical methods. In: Hybrid Systems: Computation and Control, LNCS 1790, pp. 130–144. Springer (2000)
18. Ramdani, N., Meslem, N., Candau, Y.: Reachability analysis of uncertain nonlinear systems using guaranteed set integration. In: Proc. of the 17th IFAC World Congress, pp. 8972–8977 (2008)
19. Ramdani, N., Meslem, N., Candau, Y.: Reachability of uncertain nonlinear systems using a nonlinear hybridization. In: Hybrid Systems: Computation and Control, LNCS 4981, pp. 415–428. Springer (2008)
20. Nedialkov, N.S., Jackson, K.R.: Perspectives on Enclosure Methods, chap. A New Perspective on the Wrapping Effect in Interval Methods for Initial Value Problems for Ordinary Differential Equations, pp. 219–264. Springer-Verlag (2001)
21. Krasnochtanova, I., Rauh, A., Kletting, M., Aschemann, H., Hofer, E.P., Schoop, K.M.: Interval methods as a simulation tool for the dynamics of biological wastewater treatment processes with parameter uncertainties. Applied Mathematical Modeling **34**(3), 744–762 (2010)
22. Rauh, A., Auer, E., Hofer, E.P.: Valencia-ivp: A comparison with other initial value problem solvers. In: CD-Proc. of the 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics. IEEE Computer Society (2007)

23. Asarin, E., Dang, T., Frehse, G., Girard, A., Le Guernic, C., Maler, O.: Recent progress in continuous and hybrid reachability analysis. In: Proc. of the 2006 IEEE Conference on Computer Aided Control Systems Design, pp. 1582–1587 (2006)
24. Prajna, S.: Barrier certificates for nonlinear model validation. Automatica **42**(1), 117–126 (2006)
25. Girard, A., Pappas, G.J.: Verification using simulation. In: Hybrid Systems: Computation and Control, LNCS 3927, pp. 272–286. Springer (2006)
26. Kapinski, J., Donzé, A., Lerda, F., Maka, H., Wagner, S., Krogh, B.H.: Control software model checking using bisimulation functions for nonlinear systems. In: Proc. of the 47th IEEE Conference on Decision and Control, pp. 4024–4029 (2008)
27. Althoff, M.: Reachability analysis and its application to the safety assessment of autonomous cars. Dissertation, TU München (2010). URL: http://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:bvb:91-diss-20100715-963752-1-4
28. Althoff, M., Le Guernic, C., Krogh, B.H.: Reachable set computation for uncertain time-varying linear systems. In: Hybrid Systems: Computation and Control (2011)
29. Lohner, R.: Perspectives on Enclosure Methods, chap. On the Ubiquity of the Wrapping Effect in the Computation of the Error Bounds, pp. 201–217. Springer (2001)
30. Dang, T.: Vérification et synthèse des systèmes hybrides. Ph.D. thesis, Institut National Polytechnique de Grenoble (2000)
31. Rump, S.M.: Fast and parallel interval arithmetic. BIT Numerical Mathematics **39**(3), 534–554 (1999)
32. Moler, C., Loan, C.V.: Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. SIAM Review **45**(1), 3–49 (2003)
33. Coxson, G.E.: Computing exact bounds on elements of an inverse interval matrix is NP-hard. Reliable Computing **5**(2), 137–142 (1999)
34. Kolev, L.V.: Outer interval solution of the eigenvalue problem under general form parametric dependencies. Reliable Computing **12**(2), 121–140 (2006)
35. Rugh, W.J.: Linear System Theory. Prentice Hall (1996)
36. Kosheleva, O., Kreinovich, V., Mayer, G., Nguyen, H.T.: Computing the cube of an interval matrix is NP-hard. In: Proc. of the ACM symposium on Applied computing, pp. 1449–1453 (2005)
37. Ahn, H.S., Chen, Y.Q., Moore, K.L.: Maximum singular value and power of an interval matrix. In: Proc. of the 2006 IEEE International Conference on Mechatronics and Automation, pp. 678–683 (2006)
38. Dyer, M., Gritzmann, P., Hufnagel, A.: On the complexity of computing mixed volumes. SIAM Journal on Computing **27**(2), 356–400 (1998)
39. Girard, A., Guernic, C.L., Maler, O.: Efficient computation of reachable sets of linear time-invariant systems with inputs. In: Hybrid Systems: Computation and Control, LNCS 3927, pp. 257–271. Springer (2006)
40. Pillage, L.T., Rohrer, R.A.: Asymptotic waveform evaluation for timing analysis. IEEE Transactions on Computer-Aided Design **9**(4), 352–366 (1990)
41. Han, Z.: Reachability analysis of continuous dynamic systems using dimension reduction and decomposition. Ph.D. thesis, Carnegie Mellon University, Electrical and Computer Engineering Department (2005)
42. Rump, S.M.: Developments in Reliable Computing, chap. INTLAB - INTerval LABoratory, pp. 77–104. Kluwer Academic Publishers (1999)
43. Dietzenbacher, E.: A limiting property for the powers of a non-negative, reducible matrix. Structural Change and Economic Dynamics **4**, 353–366 (1993)

Chapter 5
Robustness Comparison of Tracking Controllers Using Verified Integration

Marco Kletting and Felix Antritter (✉)

Abstract In this contribution we discuss a method for investigating the robustness properties of tracking controllers using verified simulation. This method allows to compare the controllers with respect to robustness against uncertainty in the parameters of the plant and in the initial conditions of measured and unmeasured states. A robustness criterion is formulated, which can be evaluated using interval methods. To illustrate the approach, we compare the robustness properties of three conceptually different flatness based tracking controllers with dynamic output feedback, which are applied to a simple example.

5.1 Introduction

Differential flatness [1–3] is a powerful tool for motion planning and trajectory tracking for linear and nonlinear systems. In particular for nonlinear systems there is a wide acceptance of this approach, which has been applied successfully to numerous problems of industrial relevance. However, a major drawback is the lack of techniques that allow to investigate the robustness of flatness based tracking controllers against, e.g., uncertainty in the plant parameters or in the measurements due to non-ideal sensors.

Roughly speaking, differential flatness of a control system is characterized by the existence of a — possibly fictitious — flat output that allows a differential pa-

Marco Kletting
Multi-Function Airborne Radars (OPES22), Cassidian Electronics,
Woerthstr. 85, 89077 Ulm, Germany
e-mail: marco.kletting@cassidian.com

Felix Antritter
Automatisierungs- und Regelungstechnik, Universität der Bundeswehr München,
Werner-Heisenberg-Weg 39, 85579 Neubiberg, Germany
e-mail: felix.antritter@unibw.de

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_5,

rameterization of the states and inputs. Based on the differential parameterization , a tracking controller for a given reference trajectory for the flat output can be designed. In general, not all system variables which are necessary to implement the tracking controller can be measured. In this case a nonlinear tracking observer as proposed in [4] can be used. These relations are discussed here in a general manner for single-input systems and are applied to a magnetic levitation system, where only the load position can be measured.

It has been shown in [5–7] that interval methods are a suitable tool for analyzing the properties of the resulting closed loop. Using interval methods, the maximum admissible range of parameter uncertainties in the plant and in the initial state is determined such that the deviation from a desired trajectory is guaranteed to remain within specified tolerances. To be more specific, subintervals of the parameter uncertainty boxes are considered for a verified integration [8, 9] over the desired time span. A subinterval is admissible if the resulting enclosures over the complete time span lie completely inside the specified tolerances for robustness. If the enclosures are completely outside the specified tolerances for at least one point of time, the corresponding subinterval is not admissible. Further splitting is required to decide about the admissibility of all remaining intervals. Not only uncertainty in the plant parameters can be considered but also uncertainties in the initial conditions and in the available measured signals can be considered. For the verified integration a solver based on Taylor models as implemented in COSY VI [8] is used.

In this paper we illustrate the approach for the comparison of the robustness of three different flatness based tracking controllers. The compared approaches are the "classical" flatness based tracking controller with exact linearization of the tracking error dynamics (see, e.g., [2]), the tracking controller with exact feedforward linearization [10] and an approach using a nonlinear feedforward together with a linear error feedback [11–13]. The proposed robustness analysis using interval methods cannot be used to obtain general results for the different controllers but it allows to evaluate their performance for a given system. Here, we used a magnetic levitation system. This is a structurally rather simple single input differentially flat system and hence simplifies the discussion. Furthermore, this system has been chosen in [10] as an example system for the illustration of the feedforward linearizing controller.

From the discussion of the robustness analysis in this paper it becomes clear that the approach can be applied to a wide class of uncertainties (e.g. sensor errors see [7]) and controllers (with and without observers, single and multi input) and it yields very explicit results on the disturbed dynamic behavior. This is in contrast to the approach presented in [14] for the feedforward linearizing controller, which allows to determine the final set of equilibria of the closed loop system in the presence of perturbations using set inversion based on the algorithm SIVIAX (see [15]), when additional restrictions on the velocity of the assigned reference trajectories are introduced.

This paper is organized as follows. In Section 5.2, the flatness based controllers to be compared are introduced and the construction of a nonlinear tracking observer is recalled. Additionally, we recall the most important properties of a linear dynamic output feedback of reduced order, which can be used for the linear tracking controller to estimate the feedback. In Section 5.3, we introduce the magnetic levitation system and present the resulting controllers. Section 5.4 describes briefly the verified integration of nonlinear uncertain systems based on Taylor models, which is required for the robustness analysis presented in Section 5.5, where additionally the robustness criterion is formulated. Simulation results are shown in Section 5.6, where we restricted to a parameter uncertainty in the right hand side of the system equations and in the initial state in order to focus the discussion on the comparison of the tracking controllers. Finally, conclusions and an outlook on future research are given in Section 5.7.

5.2 Flatness Based Controller Design

5.2.1 Differential Flatness and Feedforward Controller Design

Flatness based controller design has been introduced, e.g., in [1] (differential algebraic setting) and [2] (differential geometric setting). Various aspects of flatness are illustrated, e.g., in [3]. In this contribution the following relations for nonlinear single input systems are used, where the dependence of the relations on the parameters are stated explicitly: For a flat system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{p},\mathbf{x},u)\,, \tag{5.1}$$

with $\mathbf{x} \in \mathbb{R}^n$, $u \in \mathbb{R}$ and the parameter vector $\mathbf{p} \in \mathbb{R}^{n_p}$, differential flatness of (5.1) implies the existence of a flat output $y_f \in \mathbb{R}$, such that

$$\begin{aligned} y_f &= h_f(\mathbf{p},\mathbf{x})\,, && (5.2)\\ \mathbf{x} &= \psi_{\mathbf{x}}(\mathbf{p},y_f,\dot{y}_f,\ldots,y_f^{(n-1)})\,, && (5.3)\\ u &= \psi_u(\mathbf{p},y_f,\dot{y}_f,\ldots,y_f^{(n)}) && (5.4) \end{aligned}$$

holds, with h_f, ψ_u, $\psi_{\mathbf{x}}$ smooth at least on an open subset of $\mathbb{R}^n$ and $\mathbb{R}^{n+1}$ respectively. Introducing the new coordinates

$$\zeta = (\zeta_1,\ldots,\zeta_n) = (y_f,\dot{y}_f,\ldots,y_f^{(n-1)})\,, \tag{5.5}$$

the flat system (5.1) can be transformed via the well defined diffeomorphism

$$\zeta = \Phi(\mathbf{p},\mathbf{x}) \tag{5.6}$$

into controller normal form

$$\begin{aligned} \dot{\zeta}_i &= \zeta_{i+1}\,, \qquad i = 1,2,\ldots n-1\,,\\ \dot{\zeta}_n &= \alpha(\mathbf{p},\zeta,u)\,. \end{aligned} \tag{5.7}$$

Setting $v = y_f^{(n)}$ yields

$$u = \psi_u(\mathbf{p}, \zeta, v) \tag{5.8}$$

in view of (5.4) and (5.5). In [10] it has been shown that

$$\alpha(\mathbf{p}, \zeta, \psi_{\mathbf{u}}(\mathbf{p}, \zeta, v)) = v \tag{5.9}$$

holds and thus, by application of the feedback law (5.8), system (5.1) is diffeomorphic to the Brunovský normal form

$$\begin{aligned} \dot{\zeta}_i &= \zeta_{i+1}, \qquad i = 1,2,\dots n-1\,, \\ \dot{\zeta}_n &= v\,, \end{aligned} \tag{5.10}$$

with new input v. A (sufficiently smooth) reference trajectory $y_{f,d} : [t_0, t_0 + T] \to \mathbb{R}$ for the flat output y_f can be assigned almost arbitrarily (excluding singularities of the differential parameterization (5.3)–(5.4)). If the reference trajectory $y_{f,d}$ satisfies the boundary conditions

$$\mathbf{x}(t_0) = \psi_{\mathbf{x}}(\mathbf{p_0}, y_{f,d}(t_0), \dot{y}_{f,d}(t_0), \dots, y_{f,d}^{(n-1)}(t_0))\,, \tag{5.11}$$

then a corresponding feedforward controller that provides $y_f(t) = y_{f,d}(t)$ for $t \in [t_0, t_0 + T]$ is given by

$$u_d(t) = \psi_u(\mathbf{p_0}, y_{f,d}(t), \dot{y}_{f,d}(t), \dots, y_{f,d}^{(n)}(t))\,. \tag{5.12}$$

For (5.11) and (5.12) it has been assumed that the parameter vector $\mathbf{p}$ of the plant (5.1) matches a nominal parameter vector $\mathbf{p_0}$.

5.2.2 *Flatness Based Tracking Controller Design*

5.2.2.1 Tracking Controller with Exact Linearization

To stabilize the tracking of a given reference trajectory $y_{f,d}$ for the flat output, the tracking error e is introduced as

$$e = y_f - y_{f,d} = \zeta_1 - \zeta_{1,d}\,. \tag{5.13}$$

In view of (5.10) it follows that

$$e^{(i)} = \zeta_{i+1} - \zeta_{i+1,d}, \quad i = 0,1,\dots,n-1\,. \tag{5.14}$$

Thus, when setting the new input v in (5.10) to

$$v_{fb} = \dot{\zeta}_{n,d} - \sum_{i=0}^{n-1} \tilde{a}_i(\zeta_{i+1} - \zeta_{i+1,d}) = \dot{\zeta}_{n,d} - \sum_{i=0}^{n-1} \tilde{a}_i e^{(i)}\,, \tag{5.15}$$

the tracking error obeys the linear differential equation

$$0 = e^{(n)} + \sum_{i=0}^{n-1} \tilde{a}_i e^{(i)}, \tag{5.16}$$

which can be made stable by a suitable choice of the $\tilde{a}_i$. Substituting (5.15) into (5.8) yields the tracking controller

$$u_{el} = \psi_{\mathbf{u}}(\zeta, v_{fb}), \tag{5.17}$$

which yields an *exact linearization* of the controlled tracking error dynamics (see (5.16)).

5.2.2.2 Tracking Controller with Feedforward Linearization

It has been pointed out in [10] that the exact linearizing feedback controller can be very sensitive with respect to uncertain parameters and it has been proposed to use a so-called feedforward linearizing controller. The feedforward linearizing controller uses the feedback law

$$u_{fl} = \psi_u(\zeta_{\mathbf{d}}, v_{fl}). \tag{5.18}$$

The name of this tracking controller is motivated by the fact that the linearizing property (5.9) holds only when $\zeta = \zeta_{\mathbf{d}}$. Via the input v_{fl} a similar feedback as in the case of exact linearization is done:

$$v_{fl} = \dot{\zeta}_{n,d} - \sum_{i=0}^{n-1} \lambda_i e^{(i)}, \tag{5.19}$$

where e is defined as in (5.13). The design of the controller parameters λ_i, $i = 0, 1, \ldots, n-1$ is then carried out based on a linearization of the closed loop dynamics about the reference trajectory (see [10] for details).

5.2.2.3 Linear Tracking Controller

As a third tracking controller we consider a linear stabilizing feedback of the form

$$u_{lin} = u_d + v_{lin}, \tag{5.20}$$

where u_d is the nonlinear feedforward controller given by (5.12) and

$$v_{lin} = -\sum_{i=0}^{n-1} k_i e^{(i)}. \tag{5.21}$$

5.2.2.4 Resulting Closed Loop System

Using the inverse of the diffeomorphism (5.6), it is possible to implement all three presented tracking controllers as a time varying feedback of the original states, i.e.

$$u = \psi'_{u,fbl}(\mathbf{p_0}, \mathbf{p_c}, \mathbf{x}, y_{f,d}, \dot{y}_{f,d}, \ldots, \overset{(n)}{y}_{f,d}) = \psi''_{u,fbl}(\mathbf{p_0}, \mathbf{p_c}, \mathbf{x}, t) \;, \tag{5.22}$$

where $\psi'_{u,fbl}$ is any of the feedback laws (5.17),(5.18) or (5.20), $\mathbf{p_0}$ is the nominal system parameter vector and $\mathbf{p_c}$ contains the controller parameters (i.e. $\tilde{a}_i$, λ_i or k_i). As a consequence, the controlled system can be summarized for all three controllers as

$$\dot{x} = \mathbf{f}(\mathbf{p}, \mathbf{x}, \psi''_{u,fbl}(\mathbf{p_0}, \mathbf{p_c}, \mathbf{x}, t)) = \mathbf{f_{fbl}}(\mathbf{p}, \mathbf{p_0}, \mathbf{p_c}, \mathbf{x}, t) \tag{5.23}$$

where $\mathbf{p}$ is the vector containing the actual plant parameters and $\mathbf{p} \neq \mathbf{p_0}$ can occur due to not exactly known parameters.

It has been shown in [16] that the tracking controllers presented in Sections 5.2.2.1–5.2.2.3 yield the same linearized closed loop dynamics for the nominal system, if the parameters in (5.19) and (5.21) respectively are chosen according to

$$\lambda_i = \tilde{a}_i + a_{c_i}, \quad k_i = \frac{1}{b_c(t)}(\tilde{a}_i + a_{c_i}), \quad i = 0, 1, \ldots, n-1 \,, \tag{5.24}$$

where the time varying parameters a_{c_i} $i = 0, 1, \ldots, n-1$ and b_c are the entries of the linearization matrices of the open loop system in the coordinates (5.5). This will not be discussed in detail here but explained by means of the example in Section 5.3 (for more details see [16]).

5.2.3 Output Feedback

5.2.3.1 Tracking using a Nonlinear Tracking Observer

For the implementation of the feedback (5.22), in general, all states have to be available for measurement. If only the output

$$\mathbf{y} = \mathbf{h}(\mathbf{p}, \mathbf{x}) \tag{5.25}$$

with $\mathbf{y} \in \mathbb{R}^m$ is available for measurement, a nonlinear tracking observer with time varying observer gain $L(t)$

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{f}(\mathbf{p_0}, \hat{\mathbf{x}}, u) + \mathbf{L}(t)(\mathbf{y} - \mathbf{h}(\mathbf{p_0}, \hat{\mathbf{x}})) \\ &= \mathbf{f}(\mathbf{p_0}, \hat{\mathbf{x}}, u) + \mathbf{L}(t)(\mathbf{h}(\mathbf{p}, \mathbf{x}) - \mathbf{h}(\mathbf{p_0}, \hat{\mathbf{x}})) \\ &= \mathbf{f_{obs}}(\mathbf{p}, \mathbf{p_0}, \mathbf{x}, \hat{\mathbf{x}}, u, t) \end{aligned} \tag{5.26}$$

as proposed in [4] can be used. The observer (5.26) basically consists of a model of the plant and a feedback of the difference of the measured and the estimated output. For the model of the plant also the nominal parameter values $\mathbf{p_0}$ are used. The time varying observer gain $\mathbf{L}(t)$ is designed such that the linearization of the nominal estimation error dynamics about the reference trajectory $y_{f,d}$, given by

$$\Delta\dot{\mathbf{x}} - \Delta\dot{\hat{\mathbf{x}}} = (\mathbf{A}(t) - \mathbf{L}\mathbf{C}(t))(\Delta\hat{\mathbf{x}} - \Delta\mathbf{x}) \tag{5.27}$$

with

$$\mathbf{A}(t) = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x_d},\mathbf{u_d}}, \qquad \mathbf{C}(t) = \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\mathbf{x_d},\mathbf{u_d}} \tag{5.28}$$

are stable. For the stabilization of (5.27), i.e. of the estimation error dynamics in the vicinity of the reference trajectory $y_{f,d}$, methods for linear time varying systems as proposed in [17] can be used, which place constant eigenvalues in the time varying observer normal form. With the tracking observer (5.26), the feedback (5.22) can be estimated using the observer states $\hat{\mathbf{x}}$

$$\hat{u} = \hat{\psi}_u(\mathbf{p_0}, \mathbf{p_c}, \hat{\mathbf{x}}, t)\,. \tag{5.29}$$

Thus, the controlled system can be summarized as

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\hat{\mathbf{x}}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}(\mathbf{p}, \mathbf{x}, \hat{\psi}_u(\mathbf{p_0}, \mathbf{p_c}, \hat{\mathbf{x}}, t), t) \\ \mathbf{f_{obs}}(\mathbf{p}, \mathbf{p_0}, \mathbf{x}, \hat{\mathbf{x}}, \hat{\psi}_u(\mathbf{p_0}, \mathbf{p_c}, \hat{\mathbf{x}}, t), t) \end{bmatrix} = \mathbf{f_{fbo}}(\mathbf{p}, \mathbf{p_0}, \mathbf{p_c}, \mathbf{x}, \hat{\mathbf{x}}, t)\,. \tag{5.30}$$

The resulting structure for the controlled systems (5.30) is shown in Fig. 5.1. It can be analyzed using the methods discussed in Sections 5.4 and 5.5.

Some additional modifications for the tracking controllers with observer can be introduced that do not change the resulting structure and will be discussed in Section 5.3.

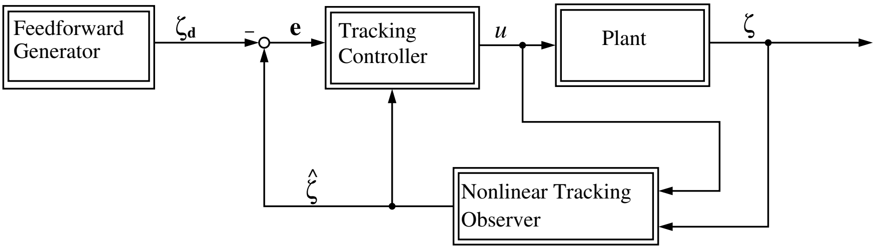


Fig. 5.1: Structure of the tracking controllers with observer

5.2.3.2 Linear Dynamic Output Feedback

Due to the linear structure of the tracking controller (5.20)–(5.21) it is possible to construct a linear dynamic output feedback which directly estimates the control law instead of estimating the states and using the estimated states to implement the feedback law. We will not go into the details of the construction of the reduced order linear dynamic output feedback, for more details see [13]. At this point we just state that it has the form

$$\dot{\xi} = \mathbf{A_O}(t)\xi + \mathbf{B_y}(t)\Delta\mathbf{y}\,, \tag{5.31}$$

$$\hat{u}_{lin} = u_d + \mathbf{C}_\xi \xi + \mathbf{C_y}(t)\Delta\mathbf{y}\,, \tag{5.32}$$

with the order $d = \dim\xi$ of the output feedback satisfying $d \leq n - m$ and that it is constructed such that the estimated control input satisfies a linear differential equation with the actual input, in other words

$$(u_{lin} - \hat{u}_{lin})^{(d)} + \sum_{i=0}^{d-1} \gamma_i (u_{lin} - \hat{u}_{lin})^{(i)} = 0\,. \tag{5.33}$$

The closed loop system becomes

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} \mathbf{f}(\mathbf{p}, \mathbf{x}, \hat{u}_{lin}(\mathbf{p_O}, \mathbf{p_c}, \xi, \Delta\mathbf{y}, t), t) \\ \mathbf{A_O}(t)\xi + \mathbf{B_y}(t)\Delta\mathbf{y} \end{bmatrix} = \mathbf{f_{do}}(\mathbf{p}, \mathbf{p_O}, \mathbf{p_c}, \mathbf{x}, \xi, t)\,. \tag{5.34}$$

The resulting structure is given in Fig. 5.2.

Of course, the linear estimation error dynamics (5.33) holds only in the vicinity of the reference trajectory since it has been assigned for the linearization. Note, however, that also the nonlinear tracking observer discussed in Section 5.2.3.1 is designed using a linearization.

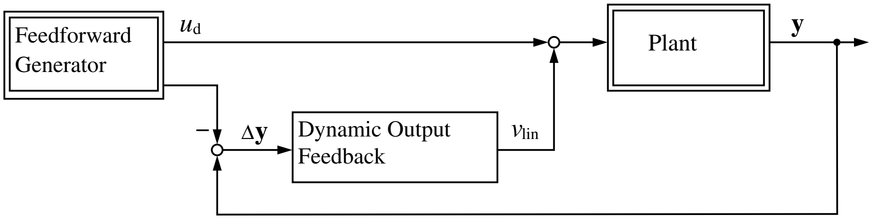


Fig. 5.2: Structure of the tracking controller with linear dynamic output feedback

5.3 Magnetic Levitation System

A simplified model of a magnetic levitation system (see Fig. 5.3) is given by [18]

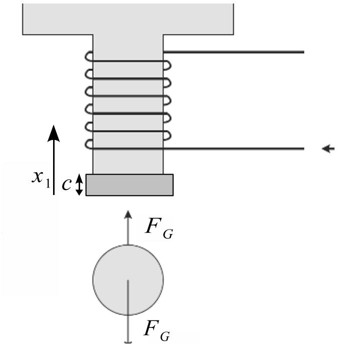


Fig. 5.3: Sketch of the magnetic levitation system

$$\begin{aligned} \dot{x}_1 &= x_2\,, \\ \dot{x}_2 &= \tfrac{k}{m}\tfrac{u^2}{(c-x_1)^2} - g\,, \end{aligned} \tag{5.35}$$

with the nominal parameters $k_0 = 58.041\frac{\mathrm{kg\,cm^3}}{\mathrm{s^2\,A^2}}$, $g_0 = 981\,\frac{\mathrm{cm}}{\mathrm{s^2}}$, $m_0 = 0.0844\,\mathrm{kg}$ and $c_0 = 0.11\,\mathrm{cm}$. The system is already given in controller normal form (5.7) and thus a flat output of (5.35) is

$$y_f = x_1\,. \tag{5.36}$$

The relations (5.3)–(5.4) can directly be derived from (5.35)

$$(x_1, x_2) = (y_f, \dot{y}_f)\,, \tag{5.37}$$

$$u = (c - y_f)\sqrt{\frac{m}{k}(\ddot{y}_f + g)}\,. \tag{5.38}$$

For the load of the levitation system a set point change is considered, i.e. a trajectory has to be planned such that the following boundary conditions are satisfied

$$\begin{aligned} (x_1(0\,\mathrm{s}), x_2(0\,\mathrm{s})) &= (-0.4\,\mathrm{cm}, 0\,\frac{\mathrm{cm}}{\mathrm{s}})\,, \\ (x_1(0.2\,\mathrm{s}), x_2(0.2\,\mathrm{s})) &= (-0.2\,\mathrm{cm}, 0\,\frac{\mathrm{cm}}{\mathrm{s}})\,. \end{aligned} \tag{5.39}$$

In view of the differential parameterization (5.37) this yields the following boundary conditions for a corresponding trajectory $y_{f,d}$ for the flat output

$$\begin{aligned} y_{f,d}(0\,\mathrm{s}) &= -0.4\,\mathrm{cm}, & \dot{y}_{f,d}(0\,\mathrm{s}) &= 0\,\frac{\mathrm{cm}}{\mathrm{s}}, \\ y_{f,d}(0.2\,\mathrm{s}) &= -0.2\,\mathrm{cm}, & \dot{y}_{f,d}(0.2\,\mathrm{s}) &= 0\,\frac{\mathrm{cm}}{\mathrm{s}}, \end{aligned} \tag{5.40}$$

which can be satisfied by assigning for $y_{f,d}$ a third order polynomial. The resulting trajectory for y_f can be seen in Fig. 5.4.

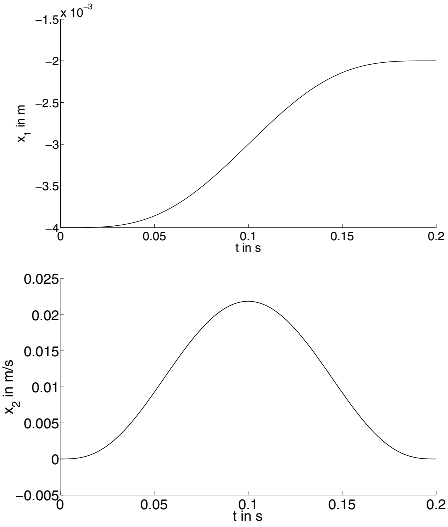


Fig. 5.4: Reference trajectory for the position x_1 and x_2 respectively

Based on the results in Section 5.2.1 a tracking controller with exact linearization of the tracking error dynamics for system (5.35) is given by (see (5.8) and (5.15))

$$u_{el} = (c - y_f)\sqrt{\frac{m}{k_c}((\ddot{y}_{f,d} - \tilde{a}_1\dot{e} - \tilde{a}_0 e) + g)}\,. \tag{5.41}$$

This controller achieves the linear tracking error dynamics

$$0 = \ddot{e} + \tilde{a}_1\dot{e} + \tilde{a}_0 e\,. \tag{5.42}$$

The linearization matrices of (5.35) in the flat coordinates clearly have the structure

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \zeta}\Big|_{\zeta_\mathbf{d},\mathbf{u_d}} = \begin{bmatrix} 0 & 1 \\ a_{c_0}(t) & 0 \end{bmatrix}, \quad \mathbf{B} = \frac{\partial \mathbf{f}}{\partial u}\Big|_{\zeta_\mathbf{d},\mathbf{u_d}} = \begin{bmatrix} 0 \\ b_c(t) \end{bmatrix}. \tag{5.43}$$

Thus, the feedforward linearizing controller which yields the same linearized closed loop dynamics as (5.41) results to (see (5.18)–(5.19) with (5.24))

$$u_{fl} = (c - y_{f,d})\sqrt{\frac{m}{k_c}((\ddot{y}_{f,d} - \tilde{a}_1\dot{e} - (a_{c_0}(t) + \tilde{a}_0)e) + g)}\,. \tag{5.44}$$

Finally, the controller with linear feedback and, again, the same linearized closed loop dynamics, results to (see (5.20)–(5.21) with (5.24))

$$u_{dor} = u_d + \frac{1}{b_c(t)}\left(-\tilde{a}_1\dot{e} - (a_{c_0}(t) + \tilde{a}_0)e\right)\,. \tag{5.45}$$

It is assumed that the flat output of (5.35) is available for measurement, i.e.

$$y = h(x) = x_1\,. \tag{5.46}$$

For the given output a nonlinear tracking observer, as discussed in Section 5.2.3.1, can be derived. It has the form

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 - l_1(t)(x_1 - \hat{x}_1)\,, \\ \dot{\hat{x}}_2 &= \tfrac{k}{m}\tfrac{u^2}{(c-\hat{x}_1)^2} - g - l_2(t)(x_1 - \hat{x}_1)\,. \end{aligned} \tag{5.47}$$

The feedback law (5.41) which stabilizes the tracking can then be estimated using the observer as

$$\hat{u}_{el} = (c - x_1)\sqrt{\frac{m}{k_c}((\ddot{y}_{f,d} - \tilde{a}_1(\hat{x}_2 - \dot{y}_{f,d}) - \tilde{a}_0(x_1 - y_{f,d})) + g)}\,, \tag{5.48}$$

where the measured output (5.46) was used to estimate x_1 and x_2 is estimated using the observer state $\hat{x}_2$. In the same manner we get for the feedback law (5.44) the estimation

$$\hat{u}_{fl} = (c - y_{f,d})\sqrt{\frac{m}{k_c}((\ddot{y}_{f,d} - \tilde{a}_1(\hat{x}_2 - \dot{y}_{f,d}) - (a_{c_0}(t) + \tilde{a}_0)(x_1 - y_{f,d})) + g)}\,. \tag{5.49}$$

Finally, the feedback law (5.45) can be estimated with the following first order linear dynamic output feedback

$$\begin{aligned} \dot{\xi} &= a_O(t)\xi + b_y(t)\Delta y\,, \\ \hat{u}_{lin} &= u_d + \xi + c_y(t)\Delta y \end{aligned} \tag{5.50}$$

with $\xi \in \mathbb{R}$.

For the robustness analysis we assume that there are constraints for the at most tolerable deviations from the reference trajectory for the controlled system which are specified in the following manner

$$|x_i(t) - x_{i,d}(t)| < \delta_i, \quad i = 1,2;\ \forall t \in [0\,\mathrm{s}, 0.2\,\mathrm{s}] \tag{5.51}$$

In Section 5.5 it will be shown how it is possible, using interval methods, to determine the admissible parameter interval $[\underline{\mathbf{p}};\overline{\mathbf{p}}]$ with $\mathbf{p} \in [\underline{\mathbf{p}};\overline{\mathbf{p}}]$ such that the tracking controllers can meet the specification (5.51). Before that, we will give a short introduction into verified simulation using Taylor models in the next section.

5.4 Verified Integration Based on Taylor Models

The controlled systems (5.30) and (5.34) respectively can be described by a set of time varying nonlinear ordinary differential equations

$$\dot{\mathbf{x}}(t) = \mathbf{f}_x(\mathbf{x}(t), \mathbf{p}(t), t), \tag{5.52}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is the state vector (including eventually the controller state for the integral error feedback) and $\mathbf{p} \in \mathbb{R}^{n_p}$ the parameter vector. The parameter vector $\mathbf{p}$ and the initial conditions $\mathbf{x}(0)$ are assumed to be uncertain with $\mathbf{p} \in [\underline{\mathbf{p}}; \overline{\mathbf{p}}]$ and $\mathbf{x}(0) \in [\underline{\mathbf{x}}(0); \overline{\mathbf{x}}(0)]$. If the parameters may vary over time within their bounds and if upper and lower bounds of the variation rate are known then

$$\dot{\mathbf{p}}(t) = \Delta\mathbf{p} \quad \text{with} \quad \Delta\mathbf{p} \in [\Delta\underline{\mathbf{p}}; \Delta\overline{\mathbf{p}}] \tag{5.53}$$

holds.

The state vector can be extended by the parameter vector according to

$$\begin{aligned} \dot{\mathbf{z}}(t) =& \mathbf{f}(\mathbf{z}(t)) \text{ with } \mathbf{z}(t) = [\mathbf{x}(t)^T, \mathbf{p}(t)^T]^T \text{ and} \\ \mathbf{f} =& \begin{bmatrix} \mathbf{f}_x(\mathbf{x}(t), \mathbf{p}(t)) \\ \Delta\mathbf{p} \end{bmatrix}, \end{aligned} \tag{5.54}$$

with $\mathbf{f} : D \mapsto \mathbb{R}^n$, $D \subset \mathbb{R}^n = \mathbb{R}^{n_x} \times \mathbb{R}^{n_p}$. Time invariant uncertain parameters are described by $\Delta\mathbf{p} = 0$.

For the robustness analysis a verified integration of the system model has to be performed. In this paper a Taylor model based integrator as implemented in COSY VI [8] is used.

Verified integration techniques like VNODE [9] are based on Taylor series expansion in time. COSY VI performs, in addition to the expansion in time, also an expansion in the initial state vector, which is in the following denoted by $\mathfrak{z}$. The domain interval vector for $\mathfrak{z}$ is given by $[\mathfrak{z}]$. The expansion point for the expansion in the initial state vector $\mathfrak{z}$ is given by $\hat{\mathfrak{z}}$ with $\hat{\mathfrak{z}} \in [\mathfrak{z}]$. The expansion point for the expansion in time is t_k. The flow of the differential equation in a given time interval $[t_k; t_{k+1}]$ is enclosed by an n-dimensional Taylor model

$$\begin{aligned} \mathbf{T}_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) &:= \mathbf{P}_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k) + \mathbf{I}_{\rho,k+1}, \\ \text{with} \quad \mathfrak{z} \in [\mathfrak{z}] \quad &\text{and} \quad t \in [t_k; t_k + 1] \ , \end{aligned} \tag{5.55}$$

where $\mathbf{P}_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ is the multivariate polynomial part of order ρ and $\mathbf{I}_{\rho,k+1}$ the remainder interval vector. Components i of $\mathbf{T}_\rho(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$ are denoted by $T_{\rho,i}(\mathfrak{z} - \hat{\mathfrak{z}}, t - t_k)$. The Taylor model at $t = t_{k+1}$ is

$$\mathbf{T}_{\rho,k+1}(\mathfrak{z}-\hat{\mathfrak{z}}) := \mathbf{P}_{\rho,k+1}(\mathfrak{z}-\hat{\mathfrak{z}}) + \mathbf{I}_{\rho,k+1} \ . \tag{5.56}$$

Components i of $\mathbf{T}_{\rho,k+1}(\mathfrak{z}-\hat{\mathfrak{z}})$ are given by $T_{\rho,i,k+1}(\mathfrak{z}-\hat{\mathfrak{z}})$.

Verified integration methods which use a single interval vector or a single parallelepiped for the state enclosure may suffer from large overestimation, especially in case of nonlinear systems. The flow representation by Taylor models makes it possible to obtain tight enclosures of non-convex sets and leads to a reduction of overestimation.

For the integration, the differential equation (5.54) is rewritten as a fixed point equation

$$\mathscr{O}(z)(t) := \mathbf{z}(t_k) + \int_{t_k}^{t} \mathbf{f}(\mathbf{z}(t'),t')dt' . \tag{5.57}$$

Applying the operator $\mathscr{O}$ to a Taylor model for the integration in the time-interval $[t_k;t_{k+1}]$ yields

$$\mathscr{O}(\mathbf{P}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\mathbf{I}_{\rho,k+1}) = \mathbf{z}(t_k) + \int_{t_k}^{t} \mathbf{f}(\mathbf{P}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t'-t_k)+\mathbf{I}_{\rho,k+1})dt' \ ,$$

where $\mathbf{z}(t_k)$ is represented by its corresponding Taylor model enclosure at $t = t_k$.

$$\mathbf{T}_{\rho,k} = \mathbf{P}_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}}) + \mathbf{I}_{\rho,k} \ . \tag{5.58}$$

This leads to

$$\begin{aligned} &\mathscr{O}(\mathbf{P}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\mathbf{I}_{\rho,k+1}) \\ &= \mathbf{P}_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}}) + \mathbf{I}_{\rho,k} + \int_{t_k}^{t} \mathbf{f}(\mathbf{P}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\mathbf{I}_{\rho,k+1})dt' \ . \end{aligned} \tag{5.59}$$

The goal for the integration from t_k to t_{k+1} is to determine a Taylor model $\mathbf{T}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)$ such that

$$\mathscr{O}(\mathbf{P}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\mathbf{I}_{\rho,k+1}) \subset \mathbf{P}_{\rho}(\mathfrak{z}-\hat{\mathfrak{z}},t-t_k)+\mathbf{I}_{\rho,k+1} \tag{5.60}$$

$\forall \mathfrak{z} \in [\mathfrak{z}]$ and $\forall t \in [t_k;t_{k+1}]$.

The polynomial part and the interval remainder are determined in two separate steps. A detailed description of these steps is given in [8, 19].

The expansion in initial states reduces the overestimation during the integration process. However, the interval remainder part remains as a source for overestimation. In order to limit the long-term growth of the remainder error and to further reduce overestimation the following strategies can be applied:

- Shrink Wrapping: here, the interval remainder is absorbed in the polynomial part [20].
- Preconditioning: the solution of ODE is studied in a different coordinate system in order to minimize long-term error growth [21].
- The domain interval vector $[\mathfrak{z}]$ can be split into subboxes and the enclosure of $\mathbf{z}(t)$ is given by a list of Taylor models [19].

5.5 Robustness Analysis of the Tracking Controllers

In this section, we discuss how the requirement (5.51) for the magnetic levitation system can be evaluated using verified integration. The goal is to determine parameter values and initial conditions of the states which are contained in the set

$$\mathbf{S}_{in} = \left\{ \begin{bmatrix} \mathbf{x_0} \\ \mathbf{p_0} \end{bmatrix} = \mathbf{z}(\mathbf{0}) \middle| \left| x_i(t) - x_{i,d}(t) \right| \leq \delta_i \forall t \in [t_0, t_0 + T],\ i = 1,2 \right\} , \tag{5.61}$$

i.e., for which it can be guaranteed that the conditions for robustness in equation (5.51) are fulfilled. On the other hand we want to determine parameter values and initial conditions which are contained in

$$\mathbf{S}_{out} = \left\{ \begin{bmatrix} \mathbf{x_0} \\ \mathbf{p_0} \end{bmatrix} = \mathbf{z}(\mathbf{0}) \middle| \exists t \in [t_0, t_0 + T], s.t. \left| x_i(t) - x_{i,d}(t) \right| > \delta_i, i \in \{1,2\} \right\} , \tag{5.62}$$

i.e., for which it can be guaranteed that the robustness conditions are not fulfilled. The δ_i, $i = 1,2$ are the allowed tolerances around the reference trajectories of the position x_1 and velocity x_2 respectively. In Fig. 5.5 examples for the evolution of the states with admissible, non-admissible and undecided parameter values of the magnetic levitation system are given when

$$\delta_1 \in [-0.2; 0.2] \cdot 10^{-3}\mathrm{m}, \quad \delta_2 \in [-0.01; 0.01]\,\frac{\mathrm{m}}{\mathrm{s}} . \tag{5.63}$$

The determination of $\mathbf{S}_{in}$ and $\mathbf{S}_{out}$ can be done by splitting $[\mathbf{p}]$ and $[\mathbf{x}(0)]$ in subboxes. Here the state vector $\mathbf{x}$, the uncertain parameters $\mathbf{p}$, the estimated state vector $\hat{\mathbf{x}}$ are combined in an extended state vector $\mathbf{z} = [\mathbf{x}^T, \hat{\mathbf{x}}^T, \mathbf{p}^T]^T$. Thus, when $[\mathbf{p}]$ and $[\mathbf{x}(0)]$ are split, the interval vector $[\mathbf{z}(0)]$ is split into subboxes

$$[\tilde{\mathbf{z}}^{(l)}(0)], l = 1,2,\ldots,L, \quad \bigcup_{l=1}^{L} [\tilde{\mathbf{z}}^{(l)}(0)] = [\mathbf{z}(0)] \ . \tag{5.64}$$

For each subbox a verified integration is performed. Here, the approach based on Taylor models as described in Section 4 is used. The algorithm is illustrated in Fig. 5.6. First an interval vector $[\tilde{\mathbf{z}}^{(l)}(0)]$ is selected for the robustness analysis, then a splitting criterion is evaluated and the selected box is split. For the split subboxes a verified integration of the system model is performed. Then, three cases have to be distinguished:

1. If, for some $t \in [t_0, t_0 + T]$, the resulting enclosure of the trajectory is completely outside the specified tolerances then the corresponding box is not admissible and can be deleted.

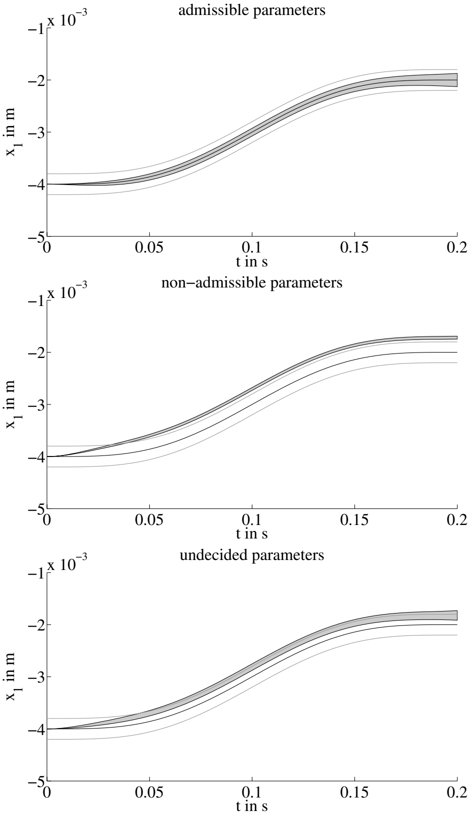


Fig. 5.5: Examples of evolutions of x_1 for admissible, non-admissible and undecided parameters

2. If, on the other hand, the resulting enclosures of the trajectory lies completely inside the tolerance for all $t \in [t_0, t_0 + T]$, then the corresponding box is admissible.
3. Subboxes which do not satisfy either of the two previous conditions have to be split further until a user given maximum number of splitting operations is reached.

Each subbox $[\tilde{\mathbf{z}}^{(l)}(0)]$ can again be expressed as a Taylor model with the unit box $[-1;1]^n$ as domain interval vector according to

$$[\tilde{\mathbf{z}}^{(l)}(0)] = \tilde{\mathbf{c}}_0^{(l)} + \tilde{\mathbf{D}}^{(l)}\mathfrak{z} \quad \text{with} \quad \mathfrak{z}_i \in [-1;1],\ i = 1, 2, \dots, n \ , \\ l = 1, 2, \dots, L \ , \tag{5.65}$$

where $\tilde{c}_0^{(l)}$ is the midpoint of $[\tilde{\mathbf{z}}^{(l)}(0)]$ and $\tilde{\mathbf{D}}^{(l)}$ is a diagonal matrix with $\tilde{D}^{(l)} = \mathrm{rad}\left([\tilde{\mathbf{z}}^{(l)}(0)]\right)$.

If $L > 1$, the most appropriate subbox for the splitting has to be selected at first. This could be the interval vector $[\tilde{\mathbf{z}}^{(l)}(0)]$ with the largest pseudo-volume. Another strategy is to calculate the pseudo volume of the interval enclosure of the Taylor model $\tilde{\mathbf{T}}^{(l)}_{\rho,k_{max}}(\mathfrak{z})$ resulting from each subbox $[\tilde{\mathbf{z}}^{(l)}(0)]$ in the last integration step of the preceding integration and select the subbox $[\tilde{\mathbf{z}}^{(l)}(0)]$ which led to the largest pseudo volume. A third selection strategy is to consider the interval remainders of the Taylor models in the last integration step, and to select the subbox which led to the interval remainder with the largest pseudo volume. After the selection of an interval vector $[\tilde{\mathbf{z}}^{(l)}(0)]$ a splitting direction has to be determined by checking the sensitivity of the Taylor model $\tilde{\mathbf{T}}^{(l)}_{\rho,k_{max}}(\mathfrak{z})$ from the selected interval vector $[\tilde{\mathbf{z}}^{(l)}(0)]$ at the last integration step of the previous integration with respect to each component $\mathfrak{z}_i$, $i = 1 \ldots n$ of the domain interval vector. The component μ of $\mathfrak{z}$ for which the Taylor model $\tilde{\mathbf{T}}^{(l)}_{\rho,k_{max}}(\mathfrak{z})$ is most sensitive is determined by the following heuristics:

First, all $w_{i,j}$

$$\begin{gathered} w_{i,j} = \mathrm{diam}([\mathfrak{z}_i]) \cdot \left| \frac{\partial T_{\rho,k+1,j}(\mathfrak{z})}{\partial \mathfrak{z}_i} \Big|_{\mathfrak{z}=\mathrm{mid}([\mathfrak{z}])} \right| , \\ i = 1,\ldots,n, \; j = 1,\ldots,n \quad , \end{gathered} \tag{5.66}$$

have to be calculated and the component μ is determined by

$$\mu = \underset{i=1\ldots n}{\arg\max} \left(\sum_{j=1}^{n} w_{i,j} \right) \quad . \tag{5.67}$$

As the interval vectors $[\tilde{\mathbf{z}}^{(l)}(0)]$ and $[\mathfrak{z}]$ are related by (5.65), the interval vector $[\tilde{\mathbf{z}}^{(l)}(0)]$ selected for splitting is also split in the component μ. Alternative methods are given in [19].

5.6 Simulation Results

For the simulation, we assume that the parameter k is not exactly known but bounded by $k \in [54; 62] \, \frac{\mathrm{kg\,cm}^3}{\mathrm{s}^2\,\mathrm{A}^2}$. Additionally, we assume that the initial state $x_2(0)$ is uncertain but bounded by $x_2(0) \in [-0.01; 0.01]$ cm. Figure 5.7 shows the parameter values in this range for which the robustness criterion (5.51) with δ_i according to (5.63) is satisfied for the controller with exact linearization of the tracking error dynamics and tracking observer. Admissible parameters are plotted in yellow, non-admissible parameters in green. Boxes with undecided parameters are depicted in red.

Figure 5.8 shows the results for the controller with feedforward linearization and tracking observer.

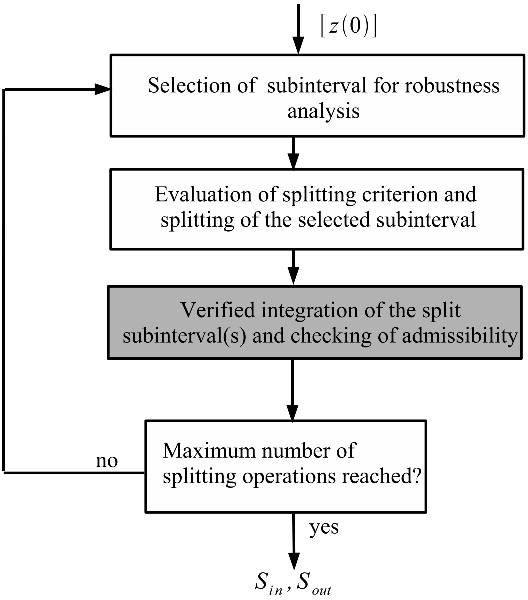

$[z(0)]$
Selection of subinterval for robustness analysis
Evaluation of splitting criterion and splitting of the selected subinterval
Verified integration of the split subinterval(s) and checking of admissibility
Maximum number of splitting operations reached?
no
yes
S_{in}, S_{out}

Fig. 5.6: Block diagram of the algorithm for the determination of the admissible parameters

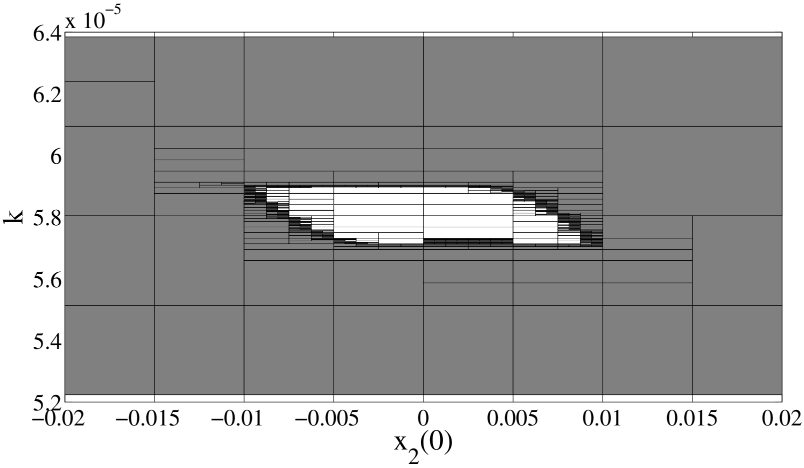


Fig. 5.7: Admissible (white), undecided (black) and non-admissible (gray) parameters for flatness based tracking controller with exact linearization and observer

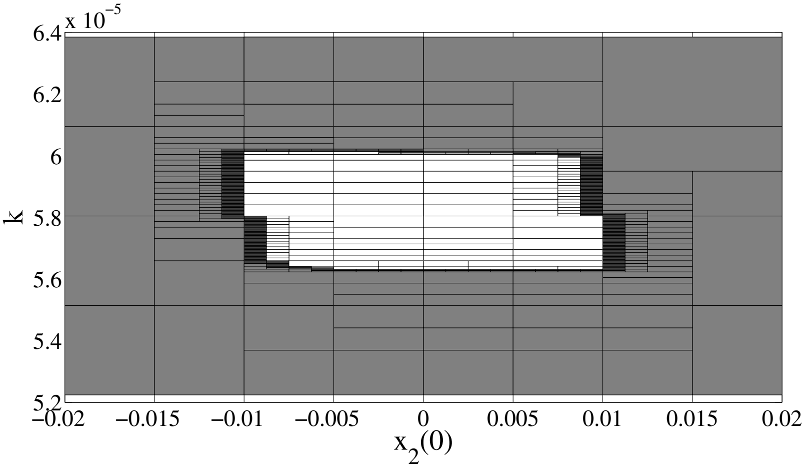


Fig. 5.8: Admissible (white), undecided (black) and non-admissible (gray) parameters for flatness based tracking controller with feedforward linearization and observer

Finally, Fig. 5.9 shows the result for the linear controller, which is estimated by the reduced order output feedback.

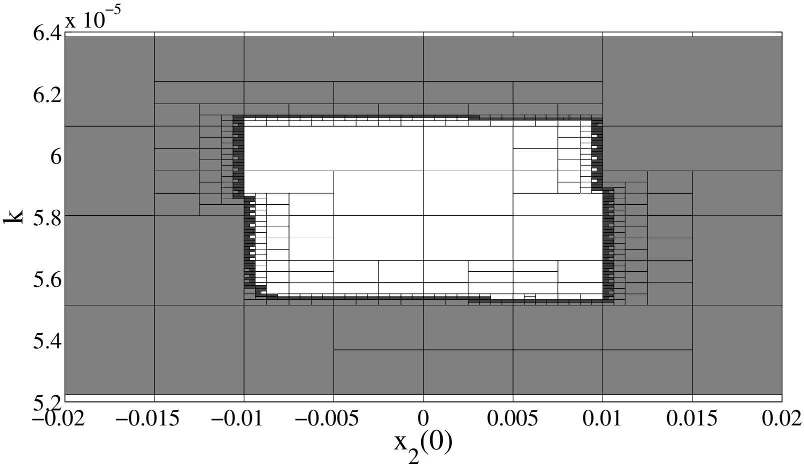


Fig. 5.9: Admissible (white), undecided (black) and non-admissible (gray) parameters for linear dynamic output feedback

The parameters $\tilde{a}$ have been chosen such that the roots of (5.42) are placed at -70. The eigenvalues of the tracking observer in observer normal form have been placed at -140. Also the root of the first order estimation error dynamics resulting from the linear dynamic output feedback (5.50) has been placed at -140. Thus, for all controllers a comparable behavior of the linearization has been used.

The feedforward linearizing controller can indeed accept a bigger parameter range than the controller with exact linearization. However, the biggest admissible parameter range is achieved by the linear controller. The better robustness properties of the linear controller result from the first order dynamic output feedback which provides, among other advantages, a more "direct" feedback of the tracking error compared to the tracking controller with tracking observer, which is of dimension two. Additionally, it is much simpler to implement the linear controller with standard commercially available controllers due its lower order and purely linear structure.

The analysis yields clear results for the considered system, but as has been mentioned already in the introduction, these results cannot be generalized to robustness properties of the different controllers for other systems.

5.7 Conclusions

In this paper the robustness of three different flatness based tracking controllers which use only output feedback for a magnetic levitation system has been analyzed using interval methods. Verified integration of subsets of the uncertain parameter and initial state led to guaranteed enclosures of the admissible sets such that the investigated controllers could satisfy the tolerances. Based on the robustness analysis, the most robust controller for the given system has been determined. Let us emphasize that such explicit results which allow to specify the maximal deviation from the desired trajectory are a new contribution for flatness based tracking controllers. Furthermore, from the discussion it has become clear that the approach can be used for rather general tracking controllers, i.e., in particular also for non flatness based tracking controllers.

It can be concluded that, together with the tools developed in earlier work [5–7], the approach provides unique possibilities for the design and verification of robust tracking controllers

The method can also be extended to other control strategies. And for the evaluation of the robustness analysis also other validated ODE solvers like VNODE [9], VALENCIA-IVP [22] or VSPODE [23] can be used.

References

1. Fliess, M., Lévine, J., Martin, P., Rouchon, P.: Flatness and defect of nonlinear systems: introductory theory and examples. Int. J. Control **61**, 1327–1361 (1995)
2. Fliess, M., Lévine, J., Martin, P., Rouchon, P.: A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems. IEEE Trans. Aut. Control **44**, 922–937 (1999)
3. Sira-Ramirez, H., Agrawal, S.K.: Differentially Flat Systems. Marcel Dekker, New York (2004)
4. Fliess, M., Rudolph, J.: Local tracking observers for flat systems. Proceedings of the Symposium on Control, Optimization and Supervision, CESA '96 IMACS Multiconference, Lille, France pp. 213–217 (1996)
5. Antritter, F., Kletting, M., Hofer, E.P.: Robustness analysis of flatness based tracking controllers using interval methods. Int. J. Control **80(5)**, 816–823 (2007)
6. Kletting, M., Antritter, F., Hofer, E.P.: Robust flatness based controller design using interval methods. In: Proceedings NOLCOS 2007. Pretoria (2007)
7. Kletting, M., Antritter, F., Hofer, E.P.: Guaranteed robust tracking with flatness based controllers applying interval methods. In: Proc. of 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006, Duisburg, Germany (2006)
8. Berz, M., Makino, K.: Verified Integration of ODEs and Flows Using Differential Algebraic Methods on High-Order Taylor Models. Reliable Computing **4**, 361–369 (1998)
9. Nedialkov, N.S., Jackson, K.R.: Methods for initial value problems for ordinary differential equations. In: R.L. U.Kulisch, A. Facius (eds.) Perspectives on Enclosure Methods, pp. 219–264. Springer-Verlag, Vienna (2001)
10. Hagenmeyer, V., Delaleau, E.: Exact feedforward linearisation based on differential flatness: The siso case. In: Nonlinear and Adaptive Control (NCN4 2001), Lecture notes in Control and Information Sciences, vol. 281, pp. 161–170. Springer, Berlin, Heidelberg (2001)
11. Deutscher, J.: A linear differential operator approach to flatness based tracking for linear and non-linear systems. Int. J. Control **76(3)**, 266–276 (2003)
12. Antritter, F., Müller, B., Deutscher, J.: Tracking control for nonlinear flat systems by linear dynamic output feedback. Proceedings NOLCOS 2004, Stuttgart (2004)
13. Antritter, F.: Tracking Control for Nonlinear Dynamics using Differential Parameterizations (PhD-Thesis). Shaker-Verlag, Aachen (2007)
14. Hagenmeyer, V., Delaleau, E.: Robustness analysis with respect to exogenous perturbations for flatness-based exact feedforward linearization. IEEE Trans. Aut. Contr. **55(3)** (2010)
15. Jaulin, L., Kieffer, M., Didrit, O., Walter, É.: Applied Interval Analysis. Springer-Verlag, London, Great Britain (2001)
16. Antritter, F.: On the relations between different flatness based design methods for tracking controllers. In: Proceedings ACC 2008, Seattle (2008)
17. Freund, E.: Zeitvariable Mehrgrößensysteme. Lecture notes in operations and mathematical science 57, Springer-Verlag, New York (1971)
18. Levine, J., Lottin, J., Ponsart, J.C.: A nonlinear approach to the control of magnetic bearings. IEEE Trans. on Control Systems Technology **4(5)**, 545 – 552 (1996)
19. Kletting, M.: Verified Methods for State and Parameter Estimators for Nonlinear Uncertain Systems with Applications in Engineering. Ph.D. thesis, Institute of Measurement, Control, and Microtechnology, University of Ulm, Germany (2009)
20. Makino, K., Berz, M.: Suppression of the wrapping effect by taylor model-based verified integrators: Long-term stabilization by shrink wrapping. International Journal of Differential Equations and Applications **10(4)**, 385–403 (2005)
21. Makino, K., Berz, M.: Suppression of the wrapping effect by taylor model-based verified integrators: Long-term stabilization by preconditioning. International Journal of Differential Equations and Applications **10(4)**, 353–384 (2005)
22. Auer, E., Rauh, A., Hofer, E.P., Luther, W.: Validated Modeling of Mechanical Systems with SmartMOBILE: Improvement of Performance by ValEncIA-IVP. In: Proc. Dagstuhl-Seminar

06021: Reliable Implementation of Real Number Algorithms: Theory and Practice, *Lecture Notes on Computer Science*, vol. 5045, pp. 1–28. Dagstuhl, Germany (2008)

23. Lin, Y., Stadtherr, M.A.: Deterministic Global Optimization for Dynamic Systems Using Interval Analysis. In: Proc. of 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006, Duisburg, Germany (2006)

Chapter 6
Probabilistic Set-Membership State Estimator

Luc Jaulin

Abstract Interval constraint propagation methods have been shown to be efficient, robust and reliable to solve difficult nonlinear bounded-error state estimation problems. However they are considered as unsuitable in a probabilistic context, where the approximation of a probability density function by a set cannot be accepted as reliable. This paper proposes a new probabilistic approach which makes it possible to use classical set-membership observers which are robust with respect to outliers. The approach is illustrated on a localization of robots in situations where there exist a large number of outliers.

6.1 Introduction

Set-membership state estimation deals with characterizing a (preferably small) set which encloses the state vector of a model from data collected on a dynamic system [29]. In the context of bounded-error estimation, the measurement errors are assumed to be bounded and characterizing the feasible set for the state vectors amounts to solve a sequence of set problems for which interval constraint propagation [26], [9] methods have been shown to be particularly efficient, even when the model is nonlinear [19], [18], [17], [22] or [13]. In a probabilistic context, set membership approaches can still be considered even if the measurement errors are not anymore described by sets, but by probability density functions. Bayesian approaches make it possible to obtain the posterior probability density function for the state vector (see, e.g., [10]) and the set to be computed becomes the minimal volume credible set [2]. This set corresponds to the minimal volume set enclosing the associated state vector with a given probability. Interval methods can still be used to characterize credible sets [14], but they are limited to small dimensional

Luc Jaulin
ENSIETA, OSM, Lab-STICC, 2 rue François Verny, 29806 Brest, France
e-mail: jaulinlu@ensieta.fr

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_6,

parameter estimation problems with few data. Recently, it has been shown [16]. [7] that it was possible to solve efficiently traditional probabilistic parameter estimation problems using interval tools. The resulting approach provides a probability associated to computed set which is fully explained with classical probabilistic tools. The main idea of the approach is to transform a probabilistic problem into a set inversion problem. Contrary to other robust Monte-Carlo based methods (such as the Ransac algorithm [12] widely used in computer vision) the resulting algorithm is deterministic and provides guaranteed results if the assumptions are satisfied (in this context, the probability of having the assumptions satisfied was assumed to be known).

This paper extends this approach to discrete-time state estimation of dynamical systems described by the following nonlinear state equations

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{f}_k(\mathbf{x}(k),\mathbf{n}(k)) \\ \mathbf{y}(k) \quad\quad = \mathbf{g}_k(\mathbf{x}(k)), \end{cases}$$

where $\mathbf{x}$ is the state vector, $\mathbf{n}$ is the state noise and k is the time. Since the evolution function $\mathbf{f}$ depends on k, this formulation encloses situations where the state equations are time dependent or when the system depends on some known inputs. In a bounded-error context (which is not exactly what is considered in this paper), $\mathbf{n}(k)$ and $\mathbf{y}(k)$ are assumed to belong to some prior feasible sets denoted by $\mathbb{N}(k)$ and $\mathbb{Y}(k)$, respectively. The sets $\mathbb{N}(k)$ are known a priori and the sets $\mathbb{Y}(k)$ are obtained from the measurement vector $\tilde{\mathbf{y}}(k)$ of the output vector $\mathbf{y}(k)$ and take into account some bounded-error noises that could corrupt the measurements. The feasible set $\mathbb{X}(k)$ corresponding to the set of all state vectors $\mathbf{x}(k)$ that are consistent with the past can be computed recursively [4] as follows

$$\mathbb{X}(k+1) = \mathbf{f}_k\left(\mathbb{X}(k) \cap \mathbf{g}_k^{-1}(\mathbb{Y}(k)),\ \ \mathbb{N}(k)\right).$$

In this formula, the operations have to be understood in a set-theoretical sense, *i.e.*,

$$\mathbf{g}_k^{-1}(\mathbb{Y}) = \{\mathbf{x} \mid \mathbf{g}_k(\mathbf{x}) \in \mathbb{Y}\}$$

and

$$\mathbf{f}_k(\mathbb{X},\mathbb{N}) = \{\mathbf{z} \mid \exists \mathbf{x} \in \mathbb{X}, \exists \mathbf{n} \in \mathbb{N}, \mathbf{z} = \mathbf{f}_k(\mathbb{X},\mathbb{N})\}.$$

Now, in practice, it may happen that some of the $\mathbf{y}(k)$, the actual values of the output vector at time k, do not belong to their corresponding sets $\mathbb{Y}(k)$. The vector $\mathbf{y}(k)$, is said to be an *inlier* if $\mathbf{y}(k) \in \mathbb{Y}(k)$ and an *outlier* otherwise. Set-membership methods have been shown to be adapted to problems involving outliers (see, *e.g.*, [27], [23], [28], [21]). The main contribution of this paper is to give a probabilistic interpretation of existing set-membership observers or more precisely to the observer presented in [15], by assuming that

- all events "$\mathbf{y}(k) \in \mathbb{Y}(k)$", $k > 0$ and the event $\mathbf{x}(0) \in \mathbb{X}(0)$ are all independent, a priori;
- the prior probability $\pi_y = \Pr(\mathbf{y}(k) \in \mathbb{Y}(k))$ is known.

The resulting methods will make it possible to use set-membership approaches to solve state estimation problems that are expressed in a probabilistic form. Note that set-membership techniques have been already been combined with probabilistic tools [20], [8], [11], [3] in order to solve estimation problems [1], but the results of the associated algorithms could not easily be explained using classical probabilistic notions.

Section 6.2 presents a set membership observer which is robust with respect to outliers. Section 6.3 provides some probabilistic properties of the observer. An illustrative application is given in Section 6.4. Section 6.5 concludes the paper.

6.2 Robust State Estimator

In a set-membership context, estimators that are robust with respect to outliers can be obtained by using the notion of *relaxed intersection* [15]. The q-relaxed intersection of m sets $\mathbb{X}_1,\dots,\mathbb{X}_m$ of $\mathbb{R}^n$ denoted by $\overset{\{q\}}{\bigcap}\mathbb{X}_i$ is the set of all $\mathbf{x}\in\mathbb{R}^n$ which belong to all $\mathbb{X}_i$'s, except q at most. Figure 6.1 illustrates this notion for $m=6$ and $q=2,3,4$. For this example, we have

$$\overset{\{0\}}{\bigcap}\mathbb{X}_i=\overset{\{1\}}{\bigcap}\mathbb{X}_i=\emptyset,\ \overset{\{5\}}{\bigcap}\mathbb{X}_i=\bigcup\mathbb{X}_i \text{ and } \overset{\{6\}}{\bigcap}\mathbb{X}_i=\mathbb{R}^2.$$

Define by induction the following notations

$$\begin{cases}\mathbf{f}_{k:k}(\mathbb{X}) & \overset{\text{def}}{=}\mathbb{X}\\ \mathbf{f}_{k_1:k_2+1}(\mathbb{X}) & \overset{\text{def}}{=}\mathbf{f}_{k_2}(\mathbf{f}_{k_1:k_2}(\mathbb{X}),\mathbb{N}(k_2)),\ k_1\leq k_2.\end{cases}$$

The set $\mathbf{f}_{k_1:k_2}(\mathbb{X})$ represents the set of all $\mathbf{x}(k_2)$, that are consistent with the fact that $\mathbf{x}(k_1)\in\mathbb{X}$. Consider the following set state estimator

$$\begin{cases}\mathbb{X}(k)=\mathbf{f}_{0:k}(\mathbb{X}(0)) \quad \text{if } k<m, \text{ (initialization step)}\\ \mathbb{X}(k)=\mathbf{f}_{k-m:k}(\mathbb{X}(k-m))\cap\\ \qquad\qquad \overset{\{q\}}{\bigcap\limits_{i\in\{1,\dots,m\}}}\mathbf{f}_{k-i:k}\circ\mathbf{g}_{k-i}^{-1}(\mathbb{Y}(k-i)) \quad \text{if } k\geq m\end{cases} \tag{6.1}$$

If we assume that (i) within any time window of length m we never have more than q outliers and that (ii) $\mathbb{X}(0)$ contains the true value for $\mathbf{x}(0)$, then $\mathbb{X}(k)$, as defined by (6.1), corresponds to the set of all feasible $\mathbf{x}(k)$ (see [15]). The principle of the observer (6.1) is illustrated by Figure 6.2 for $m=3$ and $q=1$. In this figure, double arrows are used to describe the correspondences between sets. For instance, the rightmost set corresponds to $\mathbf{f}_{k-2:k}\circ\mathbf{g}_{k-2}^{-1}(\mathbb{Y}(k-2))$ and represents the set of all $\mathbf{x}(k)$ that are consistent with the $k-2$ data set. The small gray circles are the true values of the state vectors $\mathbf{x}(k-i)$ and output vectors $\mathbf{y}(k-i)$. Note that $\mathbf{y}(k-2)$ is outside

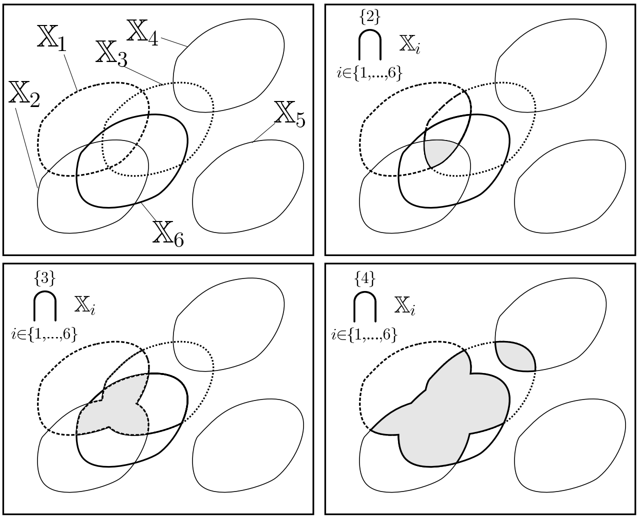

$\mathbb{X}_1$
$\mathbb{X}_4$
$\mathbb{X}_3$
$\mathbb{X}_2$
$\mathbb{X}_5$
$\mathbb{X}_6$
$\bigcap^{\{2\}}_{i\in\{1,...,6\}} \mathbb{X}_i$
$\bigcap^{\{3\}}_{i\in\{1,...,6\}} \mathbb{X}_i$
$\bigcap^{\{4\}}_{i\in\{1,...,6\}} \mathbb{X}_i$

Fig. 6.1: Illustration (in gray) of the q-relaxed intersection of the 6 sets $\mathbb{X}_1,\dots,\mathbb{X}_6$ where $q \in \{2,3,4\}$

$\mathbb{Y}(k-2)$ and is thus is an outlier, whereas $\mathbf{y}(k-1)$ and $\mathbf{y}(k-3)$ are inliers. The state estimator can efficiently be implemented using an interval constraint propagation approach which recursively computes supersets which enclose the $\mathbb{X}(k)$'s.

6.3 Probabilistic Analysis

This section provides a probabilistic interpretation of the set-membership observer presented in the previous section. We shall assume that all events "$\mathbf{y}(k) \in \mathbb{Y}(k)$", $k > 0$ and the event $\mathbf{x}(0) \in \mathbb{X}(0)$ are all independent, a priori. This assumption can be interpreted as the fact that the occurrence of an outlier at time k is independent from the past, which is close to the classical Markovian assumption. For simplicity, we shall also assume that the known prior probability $\pi_y = \Pr(\mathbf{y}(k) \in \mathbb{Y}(k))$ does not depend on k.

Proposition. Consider the following hypothesis, denoted by $\mathscr{H}_q(k_1{:}k_2)$, which states that among all $k_2 - k_1 + 1$ output vectors, $\mathbf{y}(k_1),\dots,\mathbf{y}(k_2)$, at most q of them are outlier. We have

$$\Pr\left(\mathscr{H}_q(k-m:k-1)\right) = \sum_{i=m-q}^{m} \frac{m!}{i!\,(m-i)!}\pi_y^i.\,(1-\pi_y)^{m-i}. \tag{6.2}$$

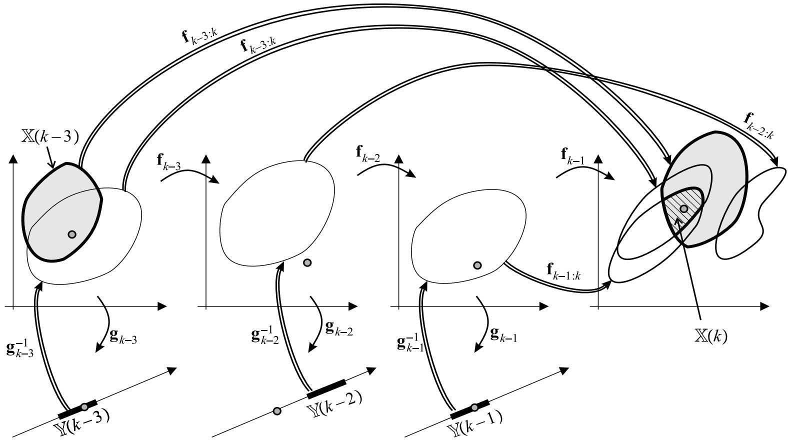


Fig. 6.2: The feasible set for the state vector $\mathbb{X}(k)$, assuming at most $q = 1$ outlier, can be defined recursively from $\mathbb{X}(k-3)$ and from the data sets $\mathbb{Y}(k-1), \mathbb{Y}(k-2), \mathbb{Y}(k-3)$.

Proof. The prior probability of having exactly i inliers among m follows a binomial distribution given by

$$\beta(i,m,\pi_y) = \frac{m!}{i!\,(m-i)!}\pi_y^i.\,(1-\pi_y)^{m-i}.$$

Thus, the probability of having at least $m-q$ inliers (or equivalently having at most q outliers) among m data is $\sum_{i=m-q}^{m} \beta(i,m,\pi_y)$. ■

Theorem. Consider the sequence of sets $\mathbb{X}(0), \mathbb{X}(1), \ldots$ built by the observer (6.1). We have

$$\Pr(\mathbf{x}(k) \in \mathbb{X}(k)) \geq \Pr(\mathbf{x}(k-m) \in \mathbb{X}(k-m)) * \sum_{i=m-q}^{m} \frac{m!\;\pi_y^i.\,(1-\pi_y)^{m-i}}{i!\,(m-i)!}.$$

Moreover, in the special case where $\mathbb{N}(k)$ are all singletons (which amounts to saying that we have no state noise) and the functions $\mathbf{f}_k$ are all injective, then the inequality becomes an equality.

Proof. Since the $m+1$ following events:

$$\mathbf{x}(k-m) \in \mathbb{X}(k-m) \text{ and } \begin{cases} \mathbf{y}(k-m) \in \mathbb{Y}(k-m) \\ \qquad\vdots \\ \mathbf{y}(k-1) \in \mathbb{Y}(k-1) \end{cases}$$

are all independent and since

$$\mathbb{X}(k) \overset{\text{def}}{=} \mathbf{f}_{k-m:k}\left(\mathbb{X}(k-m)\right) \cap \bigcap_{i\in\{1,\dots,m\}}^{\{q\}} \mathbf{f}_{k-i:k} \circ \mathbf{g}_{k-i}^{-1}\left(\mathbb{Y}(k-i)\right) \text{ if } k \geq m$$

we have the following implication

$$\left.\begin{array}{c} \mathbf{x}\,(k-m) \in \mathbb{X}(k-m) \\ \mathscr{H}_q(k-m:k-1) \end{array}\right\} \Rightarrow \mathbf{x}\,(k) \in \mathbb{X}(k). \tag{6.3}$$

Since the two events $\mathbf{x}\,(k-m) \in \mathbb{X}(k-m)$ and $\mathscr{H}_q(k-m:k-1)$, are independent, we have

$$\Pr\left(\mathbf{x}\,(k) \in \mathbb{X}(k)\right) \geq \Pr\left(\mathbf{x}\,(k-m) \in \mathbb{X}(k-m)\right) * \Pr\left(\mathscr{H}_q(k-m:k-1)\right)$$

and thus

$$\Pr\left(\mathbf{x}\,(k) \in \mathbb{X}(k)\right) \geq \Pr\left(\mathbf{x}\,(k-1) \in \mathbb{X}(k-1)\right) * \sqrt[m]{\Pr\left(\mathscr{H}_q(k-m:k-1)\right)}. \tag{6.4}$$

Assume that we have no state noise and that the $\mathbf{f}_k$ are all injective. The implication (6.3) becomes an equivalence and thus (6.4) becomes an equality. ■

6.4 Application to Localization

As an illustration, we shall now consider the problem of the localization and control of an underwater robot. The problem is similar to that presented in [15], but here, we shall add the probabilistic information. Set-membership methods have often been considered for robot localization (see, *e.g.*, [25], in the case where the problem is linear and also [5] when the robot is underwater). In situations where strong nonlinearities are involved, interval analysis has been shown to be particularly useful (see, *e.g.*, [24], [6]). Here, the approach is made more efficient by the addition of constraint propagation techniques. Assume the robot evolution is described by

$$\begin{cases} \dot{x}_1 = & x_4 \cos x_3 \\ \dot{x}_2 = & x_4 \sin x_3 \\ \dot{x}_3 = & u_2 - u_1 \\ \dot{x}_4 = & u_1 + u_2 - x_4, \end{cases}$$

where x_1, x_2 are the coordinates of the robot center, x_3 is its orientation (see Fig. 6.3) and x_4 is its speed. The inputs u_1 and u_2 are the accelerations provided by the left and right propellers, respectively. This model corresponds to an underwater robot with a constant depth (the depth regulation of the robot is assumed to be already solved and will not be considered here) and with no roll and pitch. Thus, our robot can be seen as a two-dimensional robot. The localization problem for this type of robot in the presence of outliers is similar to that treated in [24] or [19], but, in these two papers, the outliers was treated with a static manner, *i.e.*, at each k a lot of measurements were collected (24 sensors were available for the application treated). The robot pose had to be consistent with all measurements made at time k except q of them. In [15], the outliers was treated in a dynamic way, but no probability was given on the resulting set. The system can be discretized by $\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k)$ where,

$$\mathbf{f}_k \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_1 + \delta.x_4.\cos(x_3) \\ x_2 + \delta.x_4.\sin(x_3) \\ x_3 + \delta.(u_2(k) - u_1(k)) \\ x_4 + \delta.(u_1(k) + u_2(k) - x_4) \end{pmatrix}$$

and $\delta = 0.01\,\mathrm{sec}$ is the sampling time. The robot moves inside a swimming pool with a known shape (four vertical planar walls and two vertical cylinders). The robot is equipped with a sonar which measures the horizontal distance between the robot and the border of the pool following the direction pointed by the sonar. The sonar turns around itself (see Fig. 6.3) with an angular speed of $5\mathrm{rad}/\mathrm{sec}$. Denote by $\alpha(k)$ the angle between the direction of the sonar and the axis of the robot. Since the swimming pool is composed with vertical walls, the observation equation of the system has the form $d = g_k(\mathbf{x})$. Even if the functions $\mathbf{f}_k$ and g_k are strongly nonlinear, the feasible set $\mathbb{X}_k$ can efficiently be characterized using interval propagation methods. The Tchebychev center $\hat{\mathbf{x}}_k$ of $\mathbb{X}_k$ (i.e., the center of the smallest cube enclosing $\mathbb{X}_k$) is returned by our observer as an estimation of the actual state vector for the robot. This estimate is then used by the controller to compute the values $\mathbf{u}$ to be given to the propellers.

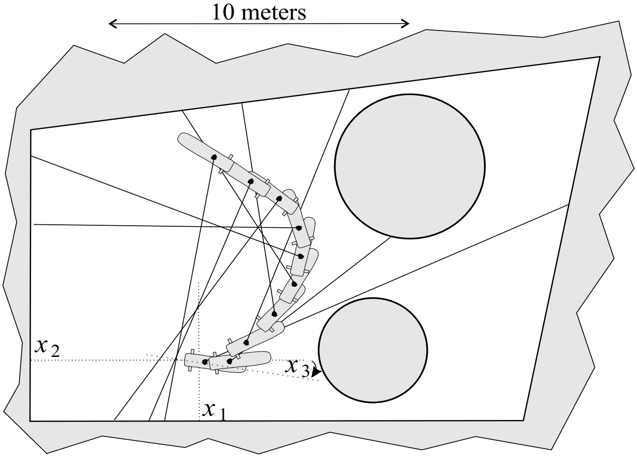


Fig. 6.3: Underwater robot moving inside a pool

Consider now a mission for the robot where three waypoints have to be reached. Once a waypoint is thought to be reached with a precision less than 0.5m, the planner sends the next waypoint, until all waypoints have been reached. The length of the sliding time window is chosen as $m = 100$, which corresponds almost to one complete turn of the sonar. The number of allowed outliers inside a time window of length m is chosen as $q = 60$. In our simulation, an outlier is generated with a probability of 0.5. In order to facilitate the visualization of the results, when an outlier is generated, the measured distance returned by the simulated robot is fixed at the unknown distance of 15m. Moreover, to the measured distance, we added a white noise with a uniform distribution inside the interval $[-0.03, 0.03]$, which corresponds to an error of ±3cm. Figure 6.4 illustrates the mission of the robot for $t \in \{3, 6, 9, 12, 15, 16.2\}$ where 16.2 sec corresponds to the duration of the mission. The black squares represent the current waypoint where the robot plans to go. The gray segments correspond the sonar distances estimated by our observer. Note that here these segments also correspond to the true distances. The small black circle represent the current position of the robot. The associated black tail represents all positions the robot had in the time interval $[t - m\delta,\ t]$. A typical emission diagram, associated to $t = 9\,\text{sec}$, is represented on Figure 6.5. The 42 outliers correspond to the gray segments. The black segments correspond to filtered distances that have been returned by our observer. The actual trajectory as well as the set-membership envelope returned by the observer are depicted on Figure 6.6. For different times t, the table below provides a lower bound for $\Pr(\mathbf{x}(k) \in \mathbb{X}(k))$ and the (unknown) number of outliers that are stored inside the current data buffer of the observer.

t(sec)	$\Pr(\mathbf{x} \in \mathbb{X})$	Outliers
3.0	≥ 0.965	58
6.0	≥ 0.932	50
9.0	≥ 0.899	42
12.0	≥ 0.869	51
15.0	≥ 0.838	51
16.2	≥ 0.827	49

As predicted by the theorem, the lower bound for $\Pr(\mathbf{x}(k) \in \mathbb{X}(k))$ decreases exponentially with respect to k. For our test-case, due to the large number q of outliers that are assumed (the number of allowed outliers inside a time window of length m was chosen as $q = 60$), this lower bound decreases fast and after few minutes it becomes almost zero. In practice, different decisions could be taken at this level.

- We want a low-cost robot for short mission and we believe that this result is sufficient.
- We want to be reliable for longer missions but we do not need to be accurate. We can thus increase the size of the $\mathbb{Y}(k)$ by increasing the error bound, we can thus assume less outliers. This will generate larger sets $\mathbb{X}(k)$, but with a higher probability to enclose $\mathbf{x}(k)$.
- We want to be both accurate and reliable. In this case, we can either add new sensors or chose more accurate sensors which generate less outliers.

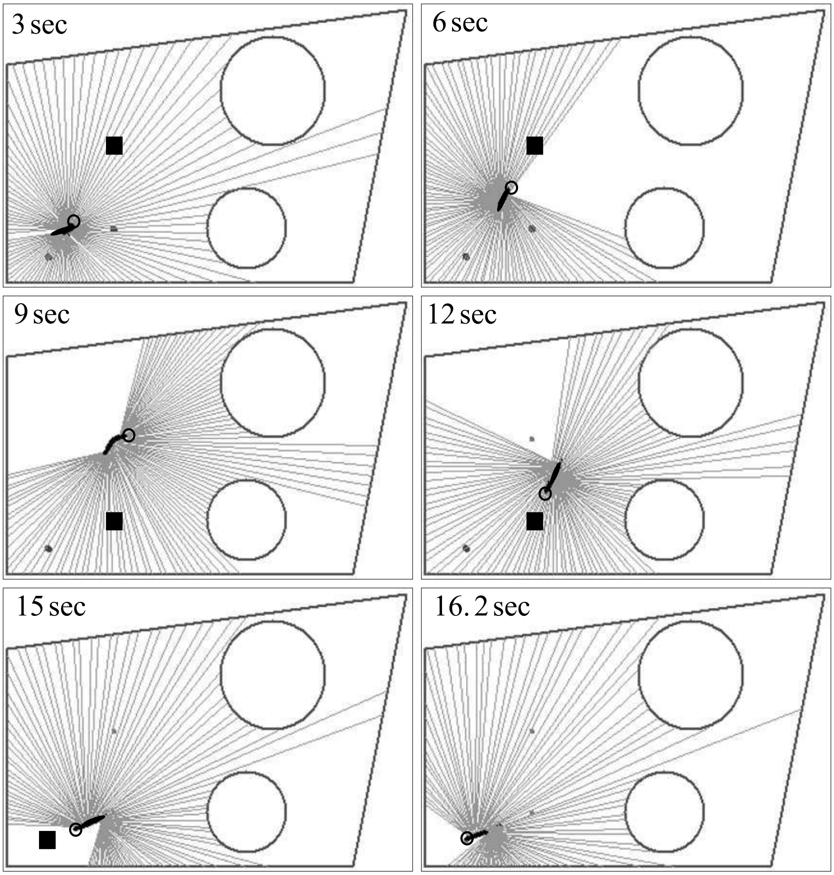


Fig. 6.4: Illustration of the robot mission for different times t

The computation time for all the mission takes less than 100 sec on classical personal computer, which makes the approach consistent with real time applications. The C++ Builder 5 source codes of this test case are available at the following address

www.ensieta.fr/jaulin/probintk.html

6.5 Conclusion

In this paper, we have proposed a new approach for state estimation which combines an interval set-membership approach with probabilities. This approach has several advantages over classical approaches. By propagating the assumptions on the possi-

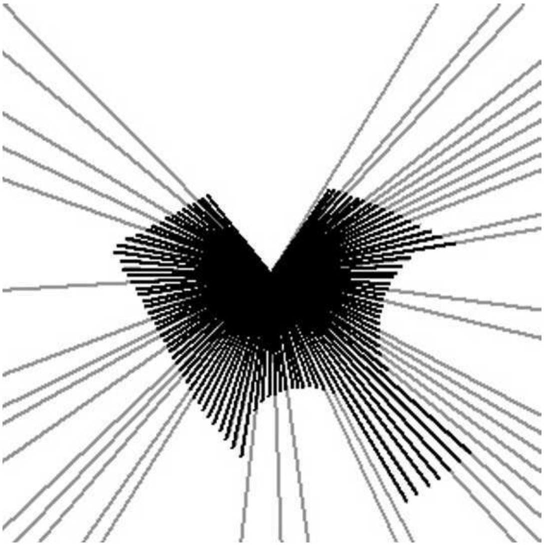


Fig. 6.5: Emission diagram at time $t = 9\,\text{sec}$

ble outliers through time, the observer has been made robust with respect to a large number of outliers. Moreover, thanks to interval analysis, the observer is able to deal with nonlinear (or non-differentiable and even noncontinuous) state equations, without linearizing or approximating them. But the remarkable property of our observer is its ability to provide a probability associated with the current set $\mathbb{X}(k)$ for the state vector $\mathbf{x}(k)$. This is new in the context of set-membership state estimation. As a consequence, the observer was able to take into account the fact that there always exists a nonzero probability that some of the set-membership assumptions are not fulfilled. The principle of the approach has been illustrated on the localization of an underwater robot where many outliers occurred during the mission.

As illustrated by the test-case, the main limitation of the proposed approach is that the lower bound for the probability that the computed set $\mathbb{X}(k)$ contains the state vector, decreases exponentially with respect to k. As a consequence, the approach for state estimation cannot be used for long missions, when the the sensors generate a large number of outliers and when a good accuracy is required to control the system.

References

1. Abdallah, F., Gning, A., Bonnifait, P.: Box particle filtering for nonlinear state estimation using interval analysis. Automatica **44**(3), 807–815 (2008)

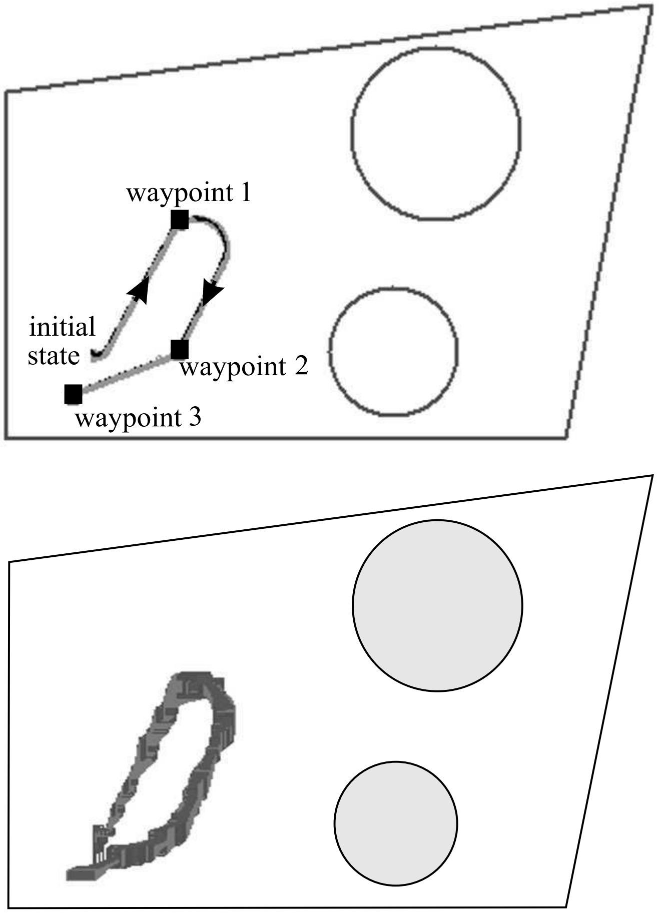


Fig. 6.6: Actual trajectory of the robot and the corresponding envelope

2. Berger, J.: Statistical Decision Theory and Bayesian Analysis, *2nd edition*. Springer-Verlag, New York, NY (1985)
3. Berleant, D., Xie, L., Zhang, J.: Statool: a tool for distribution envelope determination (denv), an interval-based algorithm for arithmetic on random variables. Reliable Computing **9**(2), 91–108 (2003)
4. Bertsekas, D.P., Rhodes, I.B.: Recursive state estimation for a set-membership description of uncertainty. "IEEE Transactions on Automatic Control" **16**(2), 117–128 (1971)
5. Caiti, A., Garulli, A., Livide, F., Prattichizzo, D.: Set-membership acoustic tracking of autonomous underwater vehicles. Acta Acustica united with Acustica **5**(88), 648–652 (2002)
6. Clrentin, A., Delafosse, M., Delahoche, L., Marhic, B., Jolly-Desodt, A.: Uncertainty and imprecision modeling for the mobile robot localization problem. Autonomous Robots **24**(3), 1573–7527 (2008)
7. Drevelle, V., Bonnifait, P.: High integrity gnss location zone characterization using interval analysis. In: ION GNSS (2009)

8. Dubois, D., Prade, H.: Random sets and fuzzy interval analysis. Fuzzy Sets and Systems **42**(1), 87–101 (1976)
9. van Emden, M.: Algorithmic power from declarative use of redundant constraints. Constraints **4**(4), 363–381 (1999)
10. Eykhoff, P.: System Identification, Parameter and State Estimation. John Wiley, London (1974)
11. Ferson, S.: RAMAS Risk Calc 4.0: Risk Assessment with Uncertain Numbers. CRC Press, Boca Raton, Florida
12. Forsyth, D., Ponce, J.: Computer Vision, a modern approach. Prentice Hall (2003)
13. Gning, A., Bonnifait, P.: Constraints propagation techniques on intervals for a guaranteed localization using redundant data. Automatica **42**(7), 1167–1175 (2006)
14. Jaulin, L.: Computing minimal-volume credible sets using interval analysis; application to bayesian estimation. IEEE Trans. on Signal Processing **54**(9), 3632–3636 (2006)
15. Jaulin, L.: Robust set membership state estimation ; application to underwater robotics. Automatica **45**(1), 202–206 (2009)
16. Jaulin, L.: Probabilistic set-membership approach for robust regression. Journal of Statistical Theory and Practice **4**(1) (2010)
17. Jaulin, L., Kieffer, M., Braems, I., Walter, E.: Guaranteed nonlinear estimation using constraint propagation on sets. International Journal of Control **74**(18), 1772–1782 (2001)
18. Jaulin, L., Kieffer, M., Didrit, O., Walter, E.: Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics. Springer-Verlag, London (2001)
19. Kieffer, M., Jaulin, L., Walter, E., Meizel, D.: Robust autonomous robot localization using interval analysis. Reliable Computing **6**(3), 337–362 (2000)
20. Kreinovich, V., Dimuro, G., da Rocha Costa, A.C.: Probabilities, intervals, what next? extension of interval computations to situations with partial information about probabilities. In: 10th IMEKO TC7 International symposium (2004)
21. Kreinovich, V., Longpr, L., Patangay, P., Ferson, S., Ginzburg, L.: Outlier detection under interval uncertainty: Algorithmic solvability and computational complexity. In: I. Lirkov, S. Margenov, J. Wasniewski, P. Yalamov (eds.) Large-Scale Scientific Computing, Proceedings of the 4th International Conference LSSC'2003 (2003)
22. Lagrange, S., Jaulin, L., Vigneron, V., Jutten, C.: Nonlinear blind parameter estimation. IEEE TAC **53**(4), 834–838 (2008)
23. Lahanier, H., Walter, E., Gomeni, R.: OMNE: a new robust membership-set estimator for the parameters of nonlinear models. Journal of Pharmacokinetics and Biopharmaceutics **15**, 203–219 (1987)
24. Meizel, D., Lvque, O., Jaulin, L., Walter, E.: Initial localization by set inversion. IEEE transactions on robotics and Automation **18**(6), 966–971 (2002)
25. Meizel, D., Preciado-Ruiz, A., Halbwachs, E.: Estimation of mobile robot localization: geometric approaches. In: M. Milanese, J. Norton, H. Piet-Lahanier, E. Walter (eds.) Bounding Approaches to System Identification, pp. 463–489. Plenum Press, New York, NY (1996)
26. Moore, R.E.: Methods and Applications of Interval Analysis. SIAM, Philadelphia, PA (1979)
27. Norton, J., Verez, S.: Outliers in bound-based state estimation and identification. Circuits and Systems **1**, 790–793 (1993)
28. Pronzato, L., Walter, E.: Robustness to outliers of bounded-error estimators and consequences on experiment design. In: M. Milanese, J. Norton, H. Piet-Lahanier, E. Walter (eds.) Bounding Approaches to System Identification, pp. 199–212. Plenum, New York (1996)
29. Walter, E., Pronzato, L.: Identification of Parametric Models from Experimental Data. Springer-Verlag, London, UK (1997)

Chapter 7
Verified Global Optimization for Estimating the Parameters of Nonlinear Models

Michel Kieffer (✉), Mihály Csaba Markót, Hermann Schichl, and Eric Walter

Abstract Nonlinear parameter estimation is usually achieved via the minimization of some possibly non-convex cost function. Interval analysis allows one to derive algorithms for the guaranteed characterization of the set of all global minimizers of such a cost function when an explicit expression for the output of the model is available or when this output is obtained via the numerical solution of a set of ordinary differential equations. However, cost functions involved in parameter estimation are usually challenging for interval techniques, if only because of multi-occurrences of the parameters in the formal expression of the cost. This paper addresses parameter estimation via the verified global optimization of quadratic cost functions. It introduces tools for the minimization of generic cost functions. When an explicit expression of the output of the parametric model is available, significant improvements may be obtained by a new box exclusion test and by careful manipulations of the quadratic cost function. When the model is described by ODEs, some of the techniques available in the previous case may still be employed, provided that sensitivity functions of the model output with respect to the parameters are available.

Michel Kieffer
Laboratoire des Signaux et Systèmes - CNRS - SUPELEC - Univ Paris-Sud, 3 rue Joliot-Curie, F-91192 Gif-sur-Yvette cedex, on leave at LTCI - CNRS - Telecom ParisTech, 46 rue Barault, F-75013 Paris, France
e-mail: michel.kieffer@lss.supelec.fr

Eric Walter
Laboratoire des Signaux et Systèmes - CNRS - SUPELEC - Univ Paris-Sud, 3 rue Joliot-Curie, F-91192 Gif-sur-Yvette cedex, France
e-mail: eric.walter@lss.supelec.fr

Mihály Csaba Markót
Fakultät für Mathematik, Universität Wien, Nordbergstr. 15, A-1090 Wien, Austria
e-mail: Mihaly.Markot@univie.ac.at

Hermann Schichl
Fakultät für Mathematik, Universität Wien, Nordbergstr. 15, A-1090 Wien, Austria
e-mail: Hermann.Schichl@univie.ac.at

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_7,

7.1 Introduction

Estimating the parameters of models from experimental data often involves the optimization of possibly non-convex cost functions. Let $\mathbf{y}(t_i)$ be the vector of all measurements collected on the system to be modeled at some time instant t_i, $i = 1, \ldots, N$. The model output $\mathbf{y}_{\mathrm{m}}(\mathbf{x}, t_i)$ at some instant t_i may consist of an explicit expression involving the vector $\mathbf{x}$ of parameters to be estimated, or it may require the solution of sets of ordinary differential equations (ODEs) containing $\mathbf{x}$ such as

$$\frac{\mathrm{d}\mathbf{z}}{\mathrm{d}t} = \mathbf{g}(\mathbf{z}, \mathbf{x}, t), \ \mathbf{z}(t_0) = \mathbf{z}_0, \tag{7.1}$$

where $\mathbf{z}$ is some state vector with initial value $\mathbf{z}_0$ at t_0 and with

$$\mathbf{y}_{\mathrm{m}}(\mathbf{x}, t) = \mathbf{h}(\mathbf{z}, \mathbf{x}, t).$$

A standard procedure for estimating $\mathbf{x}$ (see, *e.g.*, [9,38] and the references therein) is via the minimization of a cost function $f(\mathbf{x})$, which may be deduced from probabilistic assumptions on the noise affecting the measurements and on the parameters. Often, this cost function is quadratic, for instance

$$f(\mathbf{x}) = (\mathbf{y}_{\mathrm{m}}(\mathbf{x}) - \mathbf{y})^{\mathrm{T}} (\mathbf{y}_{\mathrm{m}}(\mathbf{x}) - \mathbf{y}), \tag{7.2}$$

where

$$\mathbf{y}_{\mathrm{m}}^{\mathrm{T}}(\mathbf{x}) = \left(\mathbf{y}_{\mathrm{m}}^{\mathrm{T}}(\mathbf{x}, t_1), \ldots, \mathbf{y}_{\mathrm{m}}^{\mathrm{T}}(\mathbf{x}, t_N)\right) \tag{7.3}$$

and

$$\mathbf{y}^{\mathrm{T}} = \left(\mathbf{y}^{\mathrm{T}}(t_1), \ldots, \mathbf{y}^{\mathrm{T}}(t_N)\right). \tag{7.4}$$

When $\mathbf{y}_{\mathrm{m}}(\mathbf{x})$ is linear in $\mathbf{x}$, the minimization of a quadratic $f(\mathbf{x})$ is, up to numerical stability issues, a trivial matter. Unfortunately, many models are actually nonlinear in $\mathbf{x}$, *e.g.*, knowledge-based models such as those encountered in physics, chemistry, or biology. As a consequence, $f(\mathbf{x})$ may admit in some cases several global minimizers that are all equally valid estimates. The usual local methods (such as those based on Gauss-Newton or conjugate gradient algorithms) then converge at best to a local minimizer of the cost function. Global optimization methods based on random search (for instance simulated annealing or genetic algorithms) cannot guarantee to locate all global minimizers in finite time.

Guaranteed optimization algorithms based on interval analysis [5,11,22,23,39], on the other hand, are able to derive *proven* statements about the global minimum of the cost function and associated set of global minimizers. However, cost functions involved in parameter estimation are usually challenging for interval techniques, due, *e.g.*, to multi-occurrences of the vector of parameters in the expression of the cost function [11,22]. Getting tight enclosures of cost functions over large boxes is then very difficult. This, combined with the curse of dimensionality, restrains the dimension of problems, which may be addressed using such guaranteed techniques.

The aim of this chapter is to provide some results which may help improving the efficiency of global optimization using interval techniques, especially in the case of cost functions used in parameter estimation. Tools for the guaranteed minimization of generic cost functions are first recalled in Section 7.2. Section 7.3 then focuses on techniques that significantly improve global optimization algorithms, such as constraint propagation, a new box exclusion test, and symbolic manipulations of the cost function. Such manipulations are possible when an explicit expression of the output of the parametric model is available. When the model is described by ODEs, some of the techniques introduced in Sections 7.2 and 7.3 may still be employed, provided that sensitivity functions of the model output with respect to the parameters are available, see Section 7.3.4. Improvements provided by the tools presented to the problem of parameter estimation via verified global optimization of quadratic cost functions are illustrated on a simple compartmental model in Section 7.4.

7.2 Basics of Guaranteed Optimization

This section recalls some well-known methods for guaranteed optimization that are relevant for nonlinear parameter estimation.

7.2.1 Problem Formulation

Consider the generic bound-constrained optimization problem

$$\begin{aligned} &\min f(\mathbf{x}), \\ &\text{s.t.} \quad \mathbf{x} \in [\mathbf{x}]_0, \end{aligned} \tag{7.5}$$

where $[\mathbf{x}]_0 \in \mathbb{IR}^n$ is some search box, and the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is at least twice continuously differentiable on $[\mathbf{x}]_0$. The problem of parameter estimation via minimization of some cost function may be written as (7.5), provided that a (possibly very large) initial search box $[\mathbf{x}]_0$ has been chosen.

As already mentioned in the introduction, the aim of *deterministic* global optimization is to find *rigorous* interval enclosures to *all* global minimizers and to the global minimum f^*. The most widely used scheme of interval-based global optimization methods is the branch–and–bound (B&B) technique introduced by [12,14] for discrete problems and for continuous problems by [18,35]. There have been numerous improvements, see [22] for a recent survey.

7.2.2 Why is Global Optimization for Parameter Estimation Difficult?

Assume, for the sake of simplicity, that $y(t_i)$ is scalar, so the objective function (7.2) may be rewritten as

$$f(\mathbf{x}) = \sum_{i=1}^{N} (y_{\mathrm{m}}(\mathbf{x}, t_i) - y_i)^2. \tag{7.6}$$

The cost function (7.6) consists of N squared differences between $y_{\mathrm{m}}(\mathbf{x}, t_i)$ and y_i. Each of these squares may involve several occurrences of the parameter vector $\mathbf{x}$, leading to at least N occurrences of $\mathbf{x}$ in the expression of (7.6). Getting accurate inclusion functions for f thus may be particularly challenging. Moreover, the function evaluation near the minimizers is often dominated by cancellation since the y_is and the $y_{\mathrm{m}}(\mathbf{x}, t_i)$s are often magnitudes higher than their difference. This often causes severe overestimation in the interval evaluations, which slows down branch-and-bound methods and increases the cluster effect.

7.2.3 Interval Branch–and–Bound Methods

In general, interval B&B involves the following main iteration loop (the terminology *working list* refers to the subboxes waiting for further processing, *i.e.*, those located at *open* leaves of the B&B tree):

1. Step 1: select a subbox $[\mathbf{x}] \subseteq [\mathbf{x}]_0$ from the working list;
2. Step 2: split $[\mathbf{x}]$ into subboxes $[\mathbf{x}]_i$, $i = 1, \ldots, k$;
3. Step 3: for each i run acceleration tests to eliminate $[\mathbf{x}]_i$ or parts of it that cannot contain a global minimizer;
4. Step 4: if the stopping criterion holds for the remaining part of $[\mathbf{x}]_i$, then store it in the result list R, else store the remaining part of $[\mathbf{x}]_i$ in the working list;
5. Step 5: update the best known upper bound $\tilde{f}$ of the global minimum using information acquired from $[\mathbf{x}]_i$.

Initially the working list contains only $[\mathbf{x}]_0$. The main loop is executed until the working list becomes empty. At the end of the algorithm it is ensured that enclosures of all global minimizers are in R, and the global minimum is in the interval $[\min_{[\mathbf{x}] \in R} \inf(f([\mathbf{x}])), \tilde{f}]$.

For all steps of the B&B algorithm, there exist a number of tools. Here we will focus on describing those tools that were most successful in solving nonlinear least squares problems using the `coco_gop_ex` solver (see Section 7.4.1.4). However, the methods presented in what follows are quite general and can be applied to solving all kinds of global optimization problems, see [17].

7.2.3.1 Operations on the Working List

The subbox to be subdivided is selected as the one with the smallest lower bound of interval enclosure for the cost (Moore-Skelboe rule). The boxes are split into two subboxes in a direction determined by a first order merit function given by Csendes and Ratz (rule 'B' in [24]). The stopping criterion used in Step 4 is $\mathrm{diam}(f([\mathbf{y}])) < \varepsilon$ for a pre-specified tolerance value ε.

7.2.3.2 Tools to Update $\tilde{f}$

For every subbox we compute $\sup(f([\mathbf{c}]))$ for the interval enclosure of the center $\mathbf{c}$ to update $\tilde{f}$. Furthermore, if $\sup(f([\mathbf{c}])) < \tilde{f}$, we run a local search from $\mathbf{c}$.

7.2.3.3 Tools to Prune or Erase $[\mathbf{x}]$

Most of the effort for solving a global optimization problem is spent in this phase of the solver. The effectivity of the implemented pruning or reduction techniques for subboxes is essential for the efficiency of the B&B solver.

Bound (suboptimality) test

If $\inf(V_f([\mathbf{x}])) > \tilde{f}$, then the box $[\mathbf{x}]$ cannot contain a global minimizer and may be discarded. For this test an enclosure $f([\mathbf{x}])$ of the range $V_f([\mathbf{x}])$ of the objective function on $[\mathbf{x}]$ is computed by *interval evaluation*. There are several methods for computing enclosures for the values taken by a function over a box, naive interval arithmetic being the one that requires the least effort. Naive interval arithmetic usually overestimates the range, however, and the overestimation is proportional to the radius of $[\mathbf{x}]$. This is a problem, since it often makes it impossible to eliminate boxes that are close to a global minimizer without further splitting them. Therefore, estimation methods with higher order approximation properties, *i.e.*, overestimation being $O(\mathrm{rad}([\mathbf{x}]))^p$ for $p > 1$, are needed to remove boxes close to a global minimizer. Centered forms and higher-order centered forms provide such estimates. They can be based on interval gradients or higher interval derivatives or on slopes of first or higher order (see [21, 29, 33]). Typical centered forms used to get tighter enclosures are

$$\begin{aligned}
V_f([\mathbf{x}]) &\subseteq f(\mathbf{z}) + \nabla f([\mathbf{x}])^T([\mathbf{x}] - \mathbf{z}) \\
V_f([\mathbf{x}]) &\subseteq f(\mathbf{z}) + (\nabla f(\mathbf{z})^T + \tfrac{1}{2}([\mathbf{x}] - \mathbf{z})^T \nabla^2 f([\mathbf{x}]))([\mathbf{x}] - \mathbf{z}), \\
V_f([\mathbf{x}]) &\subseteq f(\mathbf{z}) + \big(\nabla f(\mathbf{z})^T + \tfrac{1}{2}([\mathbf{x}] - \mathbf{z})^T \big(\nabla^2 f(\mathbf{z}) \\
&\qquad\qquad + \tfrac{1}{3}\sum_i \nabla^3_{i::} f([\mathbf{x}])([\mathbf{x}]_i - z_i)\big)\big)([\mathbf{x}] - \mathbf{z}),
\end{aligned}$$

where $\mathbf{z}$ is usually chosen as the center of the box $[\mathbf{x}]$, and the notation : indicates that all possible values of the index should be considered. The first has quadratic, the next cubic, and the last one quartic approximation property.

Here, the naive interval arithmetic and the first two centered forms above have been considered.

Monotonicity test

If $0 \notin V_{\nabla_i f}([\mathbf{x}])$ for some i, then $[\mathbf{x}]$ cannot contain a global minimizer in its interior. The range of the gradient over some $[\mathbf{x}]$ can also be enclosed using various methods. It can be computed by forward and backward algorithmic differentiation [28, 33], the forward evaluation giving better enclosures but taking an effort of $O(n)$ function evaluations, while the backward method produces slightly worse enclosures but requires an effort of only about two function evaluations. For both approaches the overestimation is $O(\mathrm{rad}([\mathbf{x}]))$. Centered forms can be used to get higher-order approximation properties for the gradient as well, thus increasing the effectiveness of the monotonicity test for small boxes close to a critical point. Typical centered forms are

$$V_{\nabla f}([\mathbf{x}]) \subseteq \nabla f(\mathbf{z}) + \nabla^2 f([\mathbf{x}])([\mathbf{x}] - \mathbf{z}),$$
$$V_{\nabla f}([\mathbf{x}]) \subseteq \nabla f(\mathbf{z}) + \left(\nabla^2 f(\mathbf{z}) + \tfrac{1}{2}\sum_i \nabla^3_{i::} f([\mathbf{x}])([\mathbf{x}]_i - z_i)\right)([\mathbf{x}] - \mathbf{z}),$$

where again $\mathbf{z}$ usually is chosen as the center of the box $[\mathbf{x}]$. The first centered form has quadratic and the second cubic approximation property. Here, the first centered form update of the interval gradient has only been used, whenever an interval Hessian has been computed.

Interval Newton test

A Gauss-Seidel iteration is used to solve the interval system

$$\nabla^2 f([\mathbf{x}]) \cdot ([\mathbf{x}] - \mathbf{c}) + \nabla f(\mathbf{c}) = 0, \tag{7.7}$$

with $\mathbf{c} \in [\mathbf{x}]$, to verify the uniqueness or non-existence of a stationary point in $[\mathbf{x}]$ [21]. The interval Newton test can shrink $[\mathbf{x}]$ or return a set of subboxes of $[\mathbf{x}]$ that needs to be considered further in place of $[\mathbf{x}]$.

Constraint propagation (CP)

Since every global minimizer $\mathbf{x}^*$ of (7.5) has to satisfy $f(\mathbf{x}^*) \leq \tilde{f}$, the additional constraint $f(\mathbf{x}) \leq \tilde{f}$ may be introduced. We attempt to reduce $[\mathbf{x}]$ by propagating this information back to the variables. An especially efficient method for constraint propagation is the PAID propagator [36], see Section 7.3.1. It is based on coordinating forward evaluation and backward propagation steps to reduce the bounds on all variables and intermediate nodes as much as possible.

For the least squares situation (7.6), the constraint propagator will, *e.g.*, automatically take into account the N additional constraints

$$y_{\mathrm{m}}(\mathbf{x},t_i) - y_i \in \left[-\sqrt{\tilde{f}}, \sqrt{\tilde{f}}\right], \tag{7.8}$$

which may be deduced from the fact that if $f(\mathbf{x}) \leq \tilde{f}$, then each term of the sum in (7.6) has to satisfy $(y_{\mathrm{m}}(\mathbf{x},t_i) - y_i)^2 \leq \tilde{f}$, which may be rewritten as (7.8).

Exclusion/inclusion boxes

To avoid the cluster effect [10] higher-order methods are necessary. These are usually invoked right after a new approximate local minimizer $\tilde{\mathbf{x}}$ has been detected, in order to provide a pair of boxes $\tilde{\mathbf{x}} \in [\mathbf{x}]^{\mathrm{i}} \subseteq [\mathbf{x}]^{\mathrm{e}}$. In the inclusion box $[\mathbf{x}]^{\mathrm{i}}$ uniqueness of the local optimizer is proved, along with non-existence of another local optimizer in the interior of the exclusion box $[\mathbf{x}]^{\mathrm{e}}$. This box $[\mathbf{x}]^{\mathrm{e}}$ then significantly reduces the size of the result list, since boxes are pruned from the search tree, whenever they are interior to $[\mathbf{x}]^{\mathrm{e}}$. A more detailed description of this technique can be found in Section 7.3.2.

7.3 Improving the Efficiency of Guaranteed Techniques

7.3.1 The PAID Constraint Propagator

For the PAID propagator the cost function needs to be represented as a directed acyclic graph of elementary operations, called the *model DAG* in what follows. The Forward-Backward Propagation on DAGs (FBPD) algorithm is used to compute and improve enclosures of the ranges of all nodes in the DAG. Let $\mathbf{N}$ be a node that has k children $\{\mathbf{C}_i\}_{i=1}^{k}$, denoting its input variables. The elementary operation represented by $\mathbf{N}$ is a function $g : D_g \to \mathbb{R}$, where $D_g \subseteq \mathbb{R}^k$ denotes the domain of definition of g. Hence, the relationship between $\mathbf{N}$ and its children can be written as $\mathbf{N} = g(\mathbf{C}_1, \ldots, \mathbf{C}_k)$. Let $[g]$ be an inclusion function of g. The *forward evaluation* at node $\mathbf{N}$ using the inclusion function $[g]$ is defined as

$$\mathtt{FE}(\mathbf{N}, [g]) \equiv \{\mathbb{D}(\mathbf{N}) := \mathbb{D}(\mathbf{N}) \cap [g]\}, \tag{7.9}$$

where $\mathbb{D}(\mathbf{M})$ denotes the currently best known enclosure for node $\mathbf{M}$. This forward evaluation computes the enclosure of the range of a node based on the enclosures of the ranges of its children (its input variables) using an inclusion function of the elementary operation representing that node.

The *backward propagation* prunes the enclosures associated with children based on the constraint range of their parent. In other words, for each child $\mathbf{C}_i$ the backward propagation evaluates the i-th projection of the relation $\mathbf{N} = g(\mathbf{C}_1, \dots, \mathbf{C}_k)$ on the input variable represented by $\mathbf{C}_i$. We call it the i-th backward propagation at $\mathbf{N}$ and denote it by $\mathtt{BP}(\mathbf{N}, \mathbf{C}_i)$. We define the following sequence as the backward propagation at node $\mathbf{N}$

$$\mathtt{BP}(\mathbf{N}) = \{\mathtt{BP}(\mathbf{N}, \mathbf{C}_i)\}_{i=1}^{k}. \tag{7.10}$$

Although the exact projection of relations is expensive in general, an evaluation of the exact projection of elementary operations can be obtained at low cost. Indeed, assume that from the relation $\mathbf{N} = g(\mathbf{C}_1, \dots, \mathbf{C}_k)$ one can infer an equivalent relation $\mathbf{C}_i = h_i(\mathbf{N}, \{\mathbf{C}_j\}_{j=1}^{k})$ for some $i \in \{1, \dots, k\}$, where h_i is a function from $\mathbb{R}^k$ to $\mathbb{R}$. Let $[h_i]$ be an inclusion function of h_i. The i-th backward propagation can then be obtained as follows

$$\mathtt{BP}\left(\mathbf{N}, \mathbf{C}_i\right) \equiv \left\{\mathbb{D}(\mathbf{C}_i) := \mathbb{D}(\mathbf{C}_i) \cap [h_i]\left(\mathbb{D}(\mathbf{N}), \{\mathbb{D}(\mathbf{C}_j)\}_{j=1}^{k}\right)\right\}. \tag{7.11}$$

The FBPD algorithm coordinates forward and backward steps through the model DAG by a proper ordering of the nodes. The overall scheme is independent of the type of enclosure chosen at each node. Most usual are interval enclosures of the range, but interval sets, affine enclosures, and centered-form enclosures, as well as combinations of them are also possible. The complete algorithm can be found in [36].

The PAID propagator can also be used for bound and monotonicity tests, especially if it is combined with the Karush-John generator which computes a DAG representation of the first order optimality conditions. This is very efficient for general nonlinear problems with equality and inequality constraints. However, in the case of nonlinear least squares problems, the efficiency is limited, and for the parameter estimation problem considered the overall solution time is actually about 30% higher if PAID is enabled for checking the first order optimality conditions.

7.3.2 Exclusion and Inclusion Boxes

Close to a global minimizer it is usually difficult to remove subboxes generated during the splitting phase. In [10], it was shown that avoiding the cluster effect requires at least second-order methods. For very flat problems, such as nonlinear least-squares problems, close to global minimizers second order information is usually not enough. Based on [32], we have developed in [31] a third-order method that computes large exclusion boxes for optimization problems. For `coco_gop_ex` we have implemented the special case for unconstrained problems, which can be applied for optima in the interior of $[\mathbf{x}]_0$.

Let $\mathbf{z}$ be an approximate local solution of (7.5) in the interior of $[\mathbf{x}]_0$. We compute the preconditioning matrix $\mathbf{C} \approx (\nabla^2 f(\mathbf{z}))^{-1}$ as the inverse of the point Hessian at $\mathbf{z} \in [\mathbf{x}]$ for some box $[\mathbf{x}] \subseteq [\mathbf{x}]_0$. Using this we compute the following estimates (using directed rounding and interval arithmetic)

$$\begin{aligned} \overline{\mathbf{b}} &\geq |\mathbf{C}\nabla f(\mathbf{z})|, & \mathbf{B}_0 &\geq |\mathbf{C}\nabla^2 f(\mathbf{z}) - \mathbf{I}|, \\ \mathscr{B}_{ijk} &\geq \tfrac{1}{2}\left|\sum_l C_{il}\nabla^3_{ljk} f([\mathbf{x}])\right|, \end{aligned}$$

as tightly as feasible, where $|\cdot|$ denotes the componentwise absolute value. Choose some $\mathbf{v} \in \mathbb{R}^n$ with $\mathbf{v} > 0$ and set $\mathbf{w} := (\mathbf{I} - \mathbf{B}_0)\mathbf{v}$, $a_i := \sum_{j,k} v_j \mathscr{B}_{ijk} v_k$. If $D_j = w_j^2 - 4a_j\overline{b}_j > 0$ for all j, define

$$\lambda_j^{\mathrm{e}} := \frac{w_j + \sqrt{D_j}}{2a_j}, \quad \lambda_j^{\mathrm{i}} := \frac{\overline{b}_j}{a_j\lambda_j^{\mathrm{e}}}, \quad \lambda^{\mathrm{e}} := \min_j \lambda_j^{\mathrm{e}}, \quad \lambda^{\mathrm{i}} := \max_j \lambda_j^{\mathrm{i}}.$$

Theorem 7.1. *Let all estimates above be satisfied for the box* $[\mathbf{x}]$. *If now* $D_j > 0$ *for all* j *and* $\lambda^e > \lambda^i$ *then there exists a unique critical point* $\mathbf{x}^*$ *for (7.5) in the inclusion region* $[\mathbf{x}]^i := [\mathbf{z} - \lambda^i\mathbf{v}, \mathbf{z} + \lambda^i\mathbf{v}] \cap [\mathbf{x}]$. *This is the only critical point of (7.5) in the interior of the exclusion region* $[\mathbf{x}]^e := [\mathbf{z} - \lambda^e\mathbf{v}, \mathbf{z} + \lambda^e\mathbf{v}] \cap [\mathbf{x}]$.

Proof. Only a general idea of the proof is provided here, see [31] for more details. The result follows from [32, Theorem 4.3] if we set $\mathbf{f} = \nabla f(\mathbf{x})$.

In [8] and [34] it was shown that existence and uniqueness of a zero of a C^1-function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ in the box $[\mathbf{x}]$ may be proved using the Krawczyk operator

$$K_0([\mathbf{x}], \mathbf{z}) := \mathbf{z} - \mathbf{C}\mathbf{f}(\mathbf{z}) - (\mathbf{C}\nabla\mathbf{f}([\mathbf{x}]) - \mathbf{I})([\mathbf{x}] - \mathbf{z}),$$

where $\mathbf{I}$ is the identify matrix and $\mathbf{C}$ is some arbitrary matrix. More precisely, if $K_0([\mathbf{x}], \mathbf{z}) \subseteq [\mathbf{x}]$ then $[\mathbf{x}]$ contains a zero of $\mathbf{f}$. If $K_0([\mathbf{x}], \mathbf{z}) \subseteq \mathrm{int}([\mathbf{x}])$ then there is a *unique* zero in $[\mathbf{x}]$.

Instead of K_0 we can consider the second-order Krawczyk-type operator [31]

$$K([\mathbf{x}], \mathbf{z}) := \mathbf{z} - \mathbf{C}\mathbf{f} - \left(\mathbf{C}\nabla\mathbf{f}(\mathbf{z}) - \mathbf{I} + \sum_{k=1}^{n}([\mathbf{x}]_k - \mathbf{z}_k)^{\mathrm{T}}\nabla^2\mathbf{f}([\mathbf{x}])_{::k}\right)([\mathbf{x}] - \mathbf{z}). \quad (7.12)$$

Then K has the same properties as K_0 with regard to proving existence and uniqueness of zeros of $\mathbf{f}$.

A critical point $\mathbf{x}^*$ for f in (7.5) satisfies $\nabla f(\mathbf{x}^*) = 0$, so we set $\mathbf{f} = \nabla f(\mathbf{x})$. Then we prove that for every $\mathbf{z} \in [\mathbf{x}]$ and every critical point $x^* \in [\mathbf{x}]$ the deviation $\mathbf{s} := |\mathbf{x}^* - \mathbf{z}|$ satisfies

$$0 \leq \mathbf{s} \leq \left(\mathbf{B}_0 + \sum \mathbf{s}_k \mathscr{B}_{::k}(x)\right)\mathbf{s} + \overline{\mathbf{b}}.$$

This is then used to prove that for $\mathbf{u} \in \mathbb{R}^n$, $\mathbf{u} > \mathbf{0}$, with

$$\Big(\mathbf{B}_0 + \sum \mathbf{u}_k \mathscr{B}_{::k}\Big)\mathbf{u} + \overline{\mathbf{b}} < \mathbf{u} \tag{7.13}$$

the set $\{\mathbf{x} \in [\mathbf{x}] \mid |\mathbf{x} - \mathbf{z}| \le \mathbf{u}\}$ contains a unique critical point of $\mathbf{f}$. Finally, to find such an $\mathbf{u}$ we choose an arbitrary $\mathbf{v} \in \mathbb{R}^n$, $\mathbf{v} > \mathbf{0}$ and set $\mathbf{u} := \lambda \mathbf{v}$. Equation (7.13) then leads to a quadratic equation in λ for every component of $\mathbf{u}$, which in turn proves the theorem. □

The Krawczyk-type operator (7.12) takes for $\mathbf{f} = \nabla f$ the form

$$K([\mathbf{y}],\mathbf{z}) = \mathbf{z} - \mathbf{C}\nabla f(\mathbf{z}) - (\mathbf{C}\nabla^2 f(\mathbf{z}) - \mathbf{I} + \sum_{k=1}^{n} ([\mathbf{y}]_k - \mathbf{z}_k)^T \nabla^3_{::k} f([\mathbf{y}]))([\mathbf{y}] - \mathbf{z}).$$

This operator can be used after computing the inclusion/exclusion box pair in the third-order iteration $[\mathbf{y}]^{k+1} := K([\mathbf{y}]^k, \mathrm{mid}([\mathbf{y}]^k)) \cap [\mathbf{y}]^k$ with $[\mathbf{y}]^0 = [\mathbf{x}]^i$ to further shrink the size of the inclusion box. Usually, this contracts the inclusion box in a few iterations to the limit accuracy of floating point computations. Since third-order information is used in the iteration, on the final inclusion box the enclosure of the global minimum can be computed by the third order centered form

$$\begin{aligned} V_f([\mathbf{x}]) \subseteq f(\mathbf{z}) + \big(\nabla f(\mathbf{z})^T + \tfrac{1}{2}([\mathbf{x}] - \mathbf{z})^T \big(\nabla^2 f(\mathbf{z}) \\ + \tfrac{1}{3}\sum_i \nabla^3_{i::} f([\mathbf{x}])([\mathbf{x}]_i - z_i)\big)\big)([\mathbf{x}] - \mathbf{z}). \end{aligned}$$

7.3.3 Methods Requiring an Explicit Expression for the Cost Function

Overestimation is a serious problem when solving nonlinear least-squares problems. In many cases the interval range enclosures overestimate the true range by several orders of magnitude due to the structure of the functions $y_{\mathrm{m}}(\mathbf{x}, t_i)$ in (7.6) and the severe numerical cancellation in evaluating $y_{\mathrm{m}}(\mathbf{x}, t_i) - y_i$ if $\mathbf{x}$ is close to a global minimizer.

Things get even more difficult if the evaluation of y_{m} itself is already hampered by numerical cancellation. This usually is the case if y_{m} is a linear combination of exponentials, as when the ODE (7.1) is linear in the state $\mathbf{z}$.

To increase the efficiency of the solution process, therefore, it is necessary to find an expression for y_{m} that causes as little cancellation as possible. If it, *e.g.*, can be tweaked a little bit by factoring out such that it ends up as a product of univariate functions (even if they are fairly complicated and depend on more complicated parameters) then numerical cancellation will be significantly smaller, and the interval evaluations will produce less overestimation.

In view of this effect, and partially motivated by the present study, the COCONUT environment provides special tools for the optimal interval evaluations, *i.e.* evaluations with no overestimation, of one-dimensional functions and their higher-order derivatives, see [1, 27] and Section 7.4.1.4. User defined one-dimensional functions are represented as individual nodes in the model DAGs (see Section 7.3.1). If the necessary analytic properties of the functions are supplied (domain of definition, explicit derivatives up to the specified order, limit points and values, extremal points and the associated extrema, inflexion points, poles, etc.), then optimal interval evaluations can be obtained. Furthermore, inverse function evaluations on such one-dimensional nodes (*i.e.*, the evaluation of an enclosure of all $\mathbf{x}$ values for which $f(\mathbf{x}) \in [r]$ holds for a fixed $[r]$) can also be performed using a one-dimensional root finding method. Inverse function evaluations are key ingredients needed by the constraint propagation method in Section 7.3.1.

Alternatively, the COCONUT system can autodetect complex univariate functions using a symbolic analysis on the DAG and can compute the required information by automatic curve tracing and algorithmic differentiation [30]. However, the enclosures computed by automatic curve tracing are only approximately fully optimal. They are computed by univariate validated root finding of derivatives and evaluation of the functions on the enclosures of their zeros. The enclosures of the optima produced by that method are usually a few factors of the machine epsilon wide.

An example of proper reformulation of a model is provided in Section 7.4.1.

7.3.4 Without Explicit Expression for the Cost Function

Apart from the evaluation of the set of values taken by the cost f over some box $[\mathbf{x}]$, the algorithms presented in Sections 7.2.3, 7.3.1, and 7.3.2 require the evaluation of the range of derivatives of the cost up to the third order with respect to the parameters. This section shows the way these quantities may be obtained when no explicit expression of cost function is available for the case of models described by ODEs such as (7.1).

7.3.4.1 Getting Derivatives of the Cost Function

Assume, for the sake of simplicity that the cost function is given by (7.6). In this case, its gradient is

$$\nabla f(\mathbf{x}) = 2\sum_{i=1}^{N} \left(y_{\mathrm{m}}(\mathbf{x},t_i) - y_i\right) \frac{\partial y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x}}, \tag{7.14}$$

and its Hessian matrix is

$$\nabla^2 f(\mathbf{x}) = 2\sum_{i=1}^{N} \left(\frac{\partial^2 y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x}} \partial \mathbf{x}^{\mathrm{T}} + \left(y_{\mathrm{m}}(\mathbf{x},t_i) - y_i\right) \frac{\partial y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x}} \frac{\partial y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x}^{\mathrm{T}}} \right). \tag{7.15}$$

The gradient thus can be computed via the evaluation of the first-order sensitivity function of the output of the model with respect to the parameters

$$\mathbf{s}_y(\mathbf{x},t_i) = \frac{\partial y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x}}, \tag{7.16}$$

while the Hessian matrix requires also the evaluation of the second-order sensitivity function

$$\mathbf{s}_y^2(\mathbf{x},t_i) = \frac{\partial^2 y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x} \partial \mathbf{x}^{\mathrm{T}}}. \tag{7.17}$$

Assume that the model is described by (7.1) with an output at time t_i given by $y_{\mathrm{m}}(\mathbf{x},t_i) = h(\mathbf{z},\mathbf{x},t_i)$, then

$$\frac{\partial y_{\mathrm{m}}(\mathbf{x},t_i)}{\partial \mathbf{x}} = \frac{\partial \mathbf{z}^{\mathrm{T}}(\mathbf{x},t_i)}{\partial \mathbf{x}} \frac{\partial h(\mathbf{z},\mathbf{x},t_i)}{\partial \mathbf{z}} + \frac{\partial h(\mathbf{z},\mathbf{x},t_i)}{\partial \mathbf{x}}, \tag{7.18}$$

where the main difficulty comes from the evaluation of the first-order sensitivity of the state vector $\mathbf{z}$ with respect to $\mathbf{x}$

$$\mathbf{S}_z(\mathbf{x},t_i) = \frac{\partial \mathbf{z}^{\mathrm{T}}(\mathbf{x},t_i)}{\partial \mathbf{x}}. \tag{7.19}$$

Similarly, the evaluation of the Hessian matrix requires first and second-order sensitivity functions of $\mathbf{z}$ with respect to $\mathbf{x}$.

7.3.4.2 Sensitivity Functions

When the model is described by ODEs such as (7.1), the sensitivity functions of $\mathbf{z}$ with respect to $\mathbf{x}$ are obtained easily by evaluating the partial derivatives of (7.1) with respect to $\mathbf{x}$, and inverting the order of derivation to get

$$\frac{\mathrm{d}\mathbf{S}_z(\mathbf{x},t)}{\mathrm{d}t} = \frac{\partial \mathbf{g}^{\mathrm{T}}(\mathbf{z},\mathbf{x},t)}{\partial \mathbf{x}} \tag{7.20}$$

and

$$\mathbf{S}_z(\mathbf{x},t_0) = \frac{\partial \mathbf{z}_0^{\mathrm{T}}(\mathbf{x})}{\partial \mathbf{x}}. \tag{7.21}$$

Obtaining the first-order sensitivity functions of $\mathbf{z}$ with respect to $\mathbf{x}$ thus requires solving a *coupled* system of ODEs consisting of (7.1) supplemented with (7.20) and (7.21)

$$\begin{cases} \frac{\mathrm{d}\mathbf{z}}{\mathrm{d}t} = \mathbf{g}(\mathbf{z},\mathbf{x},t) \\ \frac{\mathrm{d}\mathbf{S}_z(\mathbf{x},t)}{\mathrm{d}t} = \frac{\partial \mathbf{g}^{\mathrm{T}}(\mathbf{z},\mathbf{x},t)}{\partial \mathbf{x}} \end{cases} \text{with} \begin{cases} \mathbf{z}(t_0) = \mathbf{z}_0(\mathbf{x}) \\ \mathbf{S}_z(\mathbf{x},t_0) = \frac{\partial \mathbf{z}_0(\mathbf{x})}{\partial \mathbf{x}}. \end{cases} \tag{7.22}$$

If the dimension of the initial system of ODEs is n_z, the dimension of the coupled system of ODEs (7.22) is $n_z + n_z n_x$.

Similarly, obtaining second-order sensitivity functions of the state with respect to the parameters requires solving systems of ODEs with $n_z\left(1+n_x+n_x^2\right)$ equations. Due to the increase in complexity of the systems to be solved, higher order methods described in Section 7.3.2 are quite difficult to apply when no explicit expression of the output of the model is available.

7.3.4.3 Guaranteed Numerical Integration

Several approaches may be considered to solve (7.22). Classical methods for the solution of systems of ODEs, such as Runge-Kutta, are not able to provide the range of the solutions at each t_i, $i=1,\ldots,N$ when $\mathbf{x}$ is only known to belong to some box $[\mathbf{x}]$. Guaranteed numerical integration techniques could be employed, such as `AWA` [15], `VNODE` [20], `COSY IV` [6], `VSPODE` [13], or `ValEncIA-IVP` [25,26]. The main difficulty comes from the fact that obtaining accurate enclosures for the solutions when there are uncertain parameters, as here, and uncertain initial conditions is quite difficult.

To address this issue, one may build an *extended state* $\mathbf{z}_\mathrm{e}=\left(\mathbf{z}^\mathrm{T},\mathbf{x}^\mathrm{T}\right)^\mathrm{T}$, satisfying the following extended systems of ODEs

$$\frac{\mathrm{d}\mathbf{z}_\mathrm{e}}{\mathrm{d}t}=\begin{pmatrix}\mathbf{g}\left(\mathbf{z},\mathbf{x},t\right)\\ \mathbf{0}\end{pmatrix}, \tag{7.23}$$

if the vector of parameters is constant. The initial conditions are then

$$\mathbf{z}_\mathrm{e}\left(t_0\right)=\begin{pmatrix}\mathbf{z}_0\left(\mathbf{x}\right)\\ \mathbf{x}\end{pmatrix}, \text{ with } \mathbf{x}\in[\mathbf{x}]. \tag{7.24}$$

When only the initial conditions are undetermined, but known to belong to some box, guaranteed ODE solvers such as `COSY IV`, `VSPODE`, or `ValEncIA-IVP` perform quite well, since they are evaluating a Taylor development of the solution with interval remainder, this development being made also with respect to the initial condition.

Alternatively, one may enclose the solutions of (7.22) with uncertain $\mathbf{x}\in[\mathbf{x}]$ between a coupled pair of ODEs with deterministic initial conditions using Müller's theorem [19], see also Chapter 10 in this book. Any guaranteed tool for solving ODEs may then be used to solve this system.

7.4 Example

To illustrate the efficiency of the optimization techniques presented previously, we consider the estimation of the parameters of compartmental models. These models are widely used, *e.g.*, in biology to study metabolisms [7].

A compartmental model consists of a finite set of homogeneous subsystems, called compartments, which may exchange material between them and with the outside world. The evolution of the quantity of material in the compartments is described by a set of first-order ordinary differential equations, corresponding to conservation equations, usually assumed to be linear and time-invariant. These equations can be written in the form of a state equation.

Consider for example the model described by Figure 7.1. If $\mathbf{z} = (z_1, z_2)^{\mathrm{T}}$ is the vector of the quantities of material in the two compartments, its evolution is described by the linear state equation

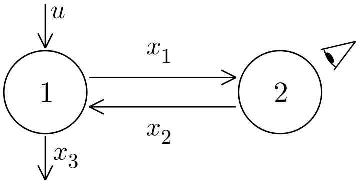


Fig. 7.1: Two-compartment model

$$\begin{cases} \dot{z}_1 = -(x_1 + x_3) z_1 + x_2 z_2, \\ \dot{z}_2 = x_1 z_1 - x_2 z_2. \end{cases} \tag{7.25}$$

Assume that the initial state is known, and such that $\mathbf{z}_0 = (1,0)^{\mathrm{T}}$, that there exists some *true* parameter value $\mathbf{x}^*$, and that only Compartment 2 is observed so that the observation equation is

$$y(t_i) = y_{\mathrm{m}}(\mathbf{x}^*, t) + b(t_i), \quad i = 1, ..., N$$

with

$$y_{\mathrm{m}}(\mathbf{x}, t) = z_2(\mathbf{x}, t_i)$$

and the $b(t_i)$'s are some noise realizations.

To generate artificial data, a two-compartment model with $\mathbf{x}^* = (0.6, 0.15, 0.35)^{\mathrm{T}}$ has been simulated. The data were then obtained by rounding the value of $z_2(t_i)$ to the nearest two-digit number for $t_i = i\Delta t$, with $\Delta t = 1$ s and $i = 1, \ldots, 15$. The initial search domain is $[\mathbf{x}]_0 = [0.01, 1]^{\times 3}$.

Parameter estimation is performed by minimizing the cost function (7.6).

7.4.1 Using an Explicit Expression for the Model Output

7.4.1.1 Original Formulation of the Problem

For the two-compartment model of Figure 7.1, one may show that the model output satisfies

$$y_{\mathrm{m}}(\mathbf{x},t_i) = \frac{x_1}{\sqrt{a(\mathbf{x})}}(e^{\lambda_1(\mathbf{x})t} - e^{\lambda_2(\mathbf{x})t}), \tag{7.26}$$

where

$$a(\mathbf{x}) = (x_3 - x_2 + x_1)^2 + 4x_1x_2, \tag{7.27}$$

$$\lambda_1(\mathbf{x}) = -(x_3 + x_2 + x_1 - \sqrt{a(\mathbf{x})})/2, \tag{7.28}$$

$$\lambda_2(\mathbf{x}) = -(x_3 + x_2 + x_1 + \sqrt{a(\mathbf{x})})/2. \tag{7.29}$$

7.4.1.2 Reformulation of the Problem

We rearrange $y_{\mathrm{m}}(\mathbf{x},t)$ as a product of univariate functions by factoring out $y_{\mathrm{m}}(\mathbf{x},t)$ as

$$y_{\mathrm{m}}(\mathbf{x},t) = \frac{x_1}{\sqrt{a(\mathbf{x})}}\left(e^{\sqrt{a(\mathbf{x})}t/2} - e^{-\sqrt{a(\mathbf{x})}t/2}\right)e^{-(x_3+x_2+x_1)t/2} \tag{7.30}$$

$$= (x_1e^{-x_1t/2})(e^{-x_2t/2})(e^{-x_3t/2})\left(\frac{e^{\sqrt{a(\mathbf{x})}t/2} - e^{-\sqrt{a(\mathbf{x})}t/2}}{\sqrt{a(\mathbf{x})}}\right). \tag{7.31}$$

The exact range of $a(\mathbf{x})$ and each parenthesized term in (7.31) can be evaluated easily with interval arithmetic (except, of course, for outward rounding) over any box in search space. Therefore, overestimation is greatly reduced when compared to the original formulation.

In the COCONUT environment we introduced the following univariate functions, on which optimal enclosures can be computed for all derivatives and the inverse function needed for the constraint propagation (see Section 7.3.1).

$$\mathrm{xexp}(v,c) := ve^{cv}, \tag{7.32}$$

$$\mathrm{hsf}(v,c) := \frac{e^{-c\sqrt{v}} - e^{c\sqrt{v}}}{\sqrt{v}} = -\frac{2}{\sqrt{v}}\sinh(c\sqrt{v}). \tag{7.33}$$

In addition we added the special quadratic node

$$\mathrm{asqr}(\mathbf{v}) := (v_3 - v_2 + v_1)^2 + 4v_1v_2. \tag{7.34}$$

with exact range evaluations up to second order and inverse function evaluations. This can be generalized for arbitrary quadratic functions, see [3,4].

With these new expressions and the exponential node (denoted here by $\exp(v,c) = e^{cv}$), $y_{\rm m}(\mathbf{x},t)$ is represented as

$$y_{\rm m}(\mathbf{x},t) = \mathrm{xexp}(x_1,c) \cdot \exp(x_2,c) \cdot \exp(x_3,c) \cdot \mathrm{hsf}(\mathrm{asqr}(\mathbf{x}),c), \tag{7.35}$$

with $c = -t/2$.

7.4.1.3 Symmetry Breaking

Note that the cost function has a x_2–x_3 permutation symmetry; this follows easily from the x_2–x_3 symmetry of $a(\mathbf{x})$ and (7.31). This symmetry was already identified in [37]. The symmetry can be broken easily to reduce the necessary computations. For this purpose we implemented a new box elimination/pruning tool in coco_gop_ex just for the present problem instance: for each box $[\mathbf{x}]$ under processing we eliminate subregions of it for which $[x_2] > [x_3]$.

Here, the symmetry, which translates in a lack of structural identifiability of the model is detected easily before performing the optimization. The automatic identification and treatment of symmetries in models represented by DAGs is an ongoing research topic in the development of the COCONUT environment.

However, if symmetries or lack of structural identifiability is not detected *a priori*, an *a posteriori* detection is possible, by looking at the set of boxes containing the candidate global optimizers, which may consists of several disconnected components. This is a definite advantage of these identification approaches based on deterministic global optimization compared to local search techniques.

7.4.1.4 Results

In the present study we used coco_gop_ex [17], the bound-constrained interval B&B solver of the COCONUT environment [1,27]. The COCONUT environment is a software platform for global optimization that provides various state-of-the-art modules that can be combined in strategies for solving global optimization problems. For bound-constrained problems the solver coco_gop_ex is provided which implements the B&B algorithm loosely described in Section 7.2.3.

For computing centered forms, *etc.*, COCONUT provides various algorithmic differentiation tools [28]. We computed $\nabla f([\mathbf{x}])$ by backward algorithmic differentiation. Second-order derivatives are computed through Hessian times vector products also with a backward evaluation scheme. The third-order derivatives are computed as follows: during the problem initialization, the DAG of the Karush-John first order necessary conditions to the problem is created; this DAG contains ∇f as a subgraph. To get third-order derivatives, the Hessian times vector product evaluator is applied on this subgraph. Alternatively, there is now a new third derivative evaluator which does not need the Karush-John conditions, but for the current test this has not yet been used.

Table 7.1: Solutions of the two-compartment model problem with different tolerance values ε. In each row, CPU is the running time in seconds, NIT is the number of B&B iterations, NBoxes is the number of small result boxes, and w is the componentwise width of the hull of the enclosures of all global minimizers (rounded to 3 nonzero decimals).

ε	CPU	NIT	NBoxes	w
10^{-2}	14	1023	1191	$[7.65, 13.2, 22.2] \cdot 10^{-2}$
10^{-3}	71	5680	5581	$[5.62, 5.89, 12.8] \cdot 10^{-2}$
10^{-4}	176	14332	2625	$[3.28, 2.20, 5.89] \cdot 10^{-2}$
10^{-5}	218	16973	1602	$[7.54, 5.62, 15.5] \cdot 10^{-3}$
10^{-6}	251	18984	790	$[1.50, 1.22, 3.12] \cdot 10^{-3}$
10^{-7}	278	21024	724	$[4.06, 2.52, 6.34] \cdot 10^{-4}$
10^{-8}	283	21377	8	$[7.79, 5.48, 13.4] \cdot 10^{-5}$
10^{-9}	284	21377	0	$[3.25, 2.21, 6.40] \cdot 10^{-12}$

For local optimization, interfaces to many different local solvers are provided by the COCONUT environment. For bound constrained problems, we use `LBFGS-B` [2].

For a detailed description of `coco_gop_ex` and the ways of synchronizing the tools used we refer to [17].

We solved the example problem on a PC with an Intel Dual-Core Mobile CPU at 1.73 GHz and with 2 GB RAM under Linux. To show how `coco_gop_ex` tackled the problem, in Table 7.1 we introduce results for different tolerance values ε from 10^{-2} to 10^{-9}. In each row of the table we gave the running time in seconds, the number of B&B iterations, the number of boxes in the result list, and the componentwise width of the search space for which we proved that it contains all global minimizers of the problem. (The latter information was computed as the componentwise hull of the elements of R^i.)

Our conclusions are the following: for larger tolerance values ($\varepsilon \geq 10^{-3}$) the problem was solved by mostly using pure splitting and first-order information. The result obtained for $\varepsilon = 10^{-3}$ with the new algorithm is similar to that of the basic interval B&B algorithm used in [37], for which the solution was obtained in about 3 hours (on a slightly slower computer). Nevertheless, `coco_gop_ex` provides the solution in just over a minute, with an approximate speedup of more than 100 times. This is due to the symbolic transformations applied to the problem (*i.e.*, the special handling of the univariate subexpressions) and the efficiency of constraint propagation. Note that a very good approximation to the global minimum is found after only a few hundred iterations. Therefore the bound test and the CP module can work efficiently with it right from the beginning of the algorithm.

Second-order tools start to work efficiently for $\varepsilon \leq 10^{-4}$, when the processed boxes reach the size of what is approximately the size of the output boxes for 10^{-3}. For instance, we found that when solving with $\varepsilon = 10^{-3}$, only around 5%

of the total amount of eliminated boxes were thrown away by the suboptimality test using second-order centered form updates, but for $\varepsilon = 10^{-4}$ this ratio was about 40%. Indeed, without using second-order information the cluster effect would have already dominated the search even for this tolerance value: when we disabled all second-order tools, we obtained the solution in around 12 minutes instead of 3, with over 64 000 result boxes! With second-order information the cluster effect is clearly avoided, as shown also by the drop in the number of output boxes from 5 600 (with $\varepsilon = 10^{-3}$) to about 2 600.

From this point the refinement of the solution with smaller and smaller tolerance values was relatively easy, *e.g.*, solving the problem with $\varepsilon = 10^{-9}$ instead of 10^{-4} took only about 110 more seconds, with continuous drops in both the number and the size of the result boxes. For $\varepsilon = 10^{-8}$ and $\varepsilon = 10^{-9}$ the boxes became small enough so that the exclusion box utility also took effect. The number 0 in the NIT column of the last row actually indicates that we have no boxes left outside the exclusion box (the componentwise widths in column w are thus the widths of the inclusion box belonging to that exclusion box). That is, at this point the solution became fully specified up to the maximal possible capabilities of our algorithm. As a summary, we found that the two-compartment example model has

- *one unique global minimizer* (apart from the x_2 – x_3 permutation symmetry), located in the interior of the (inclusion) box

$$\begin{aligned}(\ &[0.604961728242, 0.604961728246],\\ &[0.144474180373, 0.144474180376],\\ &[0.366021184203, 0.366021184210] \ )\end{aligned}$$

  (the result intervals are given outward rounded with 12 decimal digits), and
- the enclosure of the global minimum is

$$[6.72177710824, 6.72177710827] \cdot 10^{-5}.$$

The fact that the width of the enclosure of the global minimum is of the order of the machine epsilon shows that the algorithm has reached the maximal possible resolution with standard double-precision floating-point computation.

For solving this example problem, we used two tools that may not be applicable in general, namely, analytic reformulation to reduce the interval overestimation and symmetry breaking. We also solved the example problem without these two acceleration tools. Our final conclusions were precisely same as above, *i.e.*, with tolerance $\varepsilon = 10^{-9}$ we reached the tolerance of the unique optimal solution and the global optimum presented above. The performance indicators were, of course, different from those of the first run: the running time and the number or required iterations were around 6 and 4 times larger, respectively, while maximal number of result boxes (also peaking at $\varepsilon = 10^{-3}$) was around 4 times larger than in the fully accelerated method.

7.4.2 Without Using an Explicit Expression for the Model Output

7.4.2.1 Sensitivity Functions

All first-order sensitivity functions are easily derived from (7.25). These sensitivity functions are denoted $s_{ij}(\mathbf{x},t) = \partial z_i/\partial x_j$ in what follows. Sensitivity functions may be obtained by pairs, for example

$$\begin{cases} \frac{\mathrm{d}s_{11}}{\mathrm{d}t} = -z_1 - (x_1 + x_3)s_{11} + x_2 s_{21} \\ \frac{\mathrm{d}s_{21}}{\mathrm{d}t} = z_1 + x_1 s_{11} - x_2 s_{21} \end{cases} \tag{7.36}$$

with $s_{11}(0) = s_{21}(0) = 0$. However, (7.36) cannot be solved alone, as it requires to be coupled with (7.25). Thus, all first-order sensitivity function together with the system output require the solution of three coupled systems of ODEs of dimension 4.

7.4.2.2 Müller's Theorem

When outer-bounding the range of the gradient of the cost over some box $[\mathbf{x}]$, one has to evaluate the set of values taken by the sensitivity functions over $[\mathbf{x}]$. For that purpose, Müller's theorem is used to get for each coupled system of four *uncertain* ODEs defined in Section 7.4.2.1, a coupled system of eight *deterministic* ODEs. For example, to get enclosures of the state and of the sensitivity function of the state with respect to x_1, one has to solve the following system of ODEs

$$\begin{cases} \frac{\mathrm{d}\underline{z}_1}{\mathrm{d}t} = -(\overline{x}_1 + \overline{x}_3)\underline{z}_1 + \underline{x}_2\underline{z}_2 \\ \frac{\mathrm{d}\underline{z}_2}{\mathrm{d}t} = \underline{x}_1\underline{z}_1 - \overline{x}_2\underline{z}_2 \\ \frac{\mathrm{d}\overline{z}_1}{\mathrm{d}t} = -(\underline{x}_1 + \underline{x}_3)\overline{z}_1 + \overline{x}_2\overline{z}_2 \\ \frac{\mathrm{d}\overline{z}_2}{\mathrm{d}t} = \overline{x}_1\overline{z}_1 - \underline{x}_2\overline{z}_2 \\ \frac{\mathrm{d}\underline{s}_{11}}{\mathrm{d}t} = -\overline{z}_1 - (\overline{x}_1 + \overline{x}_3)\underline{s}_{11} + \underline{x}_2\underline{s}_{21} \\ \frac{\mathrm{d}\underline{s}_{21}}{\mathrm{d}t} = \underline{z}_1 + \underline{x}_1\underline{s}_{11} - \overline{x}_2\underline{s}_{21} \\ \frac{\mathrm{d}\overline{s}_{11}}{\mathrm{d}t} = -\underline{z}_1 - (\underline{x}_1 + \underline{x}_3)\overline{s}_{11} + \overline{x}_2\overline{s}_{21} \\ \frac{\mathrm{d}\overline{s}_{21}}{\mathrm{d}t} = \overline{z}_1 + \overline{x}_1\overline{s}_{11} - \underline{x}_2\overline{s}_{21} \end{cases} \tag{7.37}$$

with $\underline{z}_1(0) = \overline{z}_1(0) = 1$, $\underline{z}_2(0) = \overline{z}_2(0) = 0$, and $\underline{s}_{11}(0) = \underline{s}_{21}(0) = \overline{s}_{11}(0) = \overline{s}_{21}(0) = 0$. At any time instant t_i, when $\mathbf{x} \in [\mathbf{x}]$, an enclosure for $z_1([\mathbf{x}],t_i)$ is given by $[\underline{z}_1([\mathbf{x}],t_i), \overline{z}_1([\mathbf{x}],t_i)]$, obtained by solving (7.37). Similar enclosures are obtained for $z_2([\mathbf{x}],t_i)$, $s_{12}([\mathbf{x}],t_i)$, $s_{22}([\mathbf{x}],t_i)$, $s_{13}([\mathbf{x}],t_i)$, and $s_{23}([\mathbf{x}],t_i)$.

7.4.2.3 Results

Only first-order sensitivity functions have been used. Guaranteed numerical integration of systems such as (7.37) has been performed using VNODE-LP. Thus, only basic box elimination tests using first-order derivatives were implemented. An equivalent of the PAID contractor has been employed using centered forms for the cost function.

For a tolerance parameter $\varepsilon = 0.001$, a list of boxes is obtained whose projections onto the (x_1,x_2) plane and (x_2,x_3) plane are shown in Figure 7.2. Only boxes for which it was not possible to prove that they do not contain any global minimizer are represented, so this is an outer approximation. This result has been obtained in 3 h on a Pentium IV at 2 GHz. The set of boxes contains the solution provided in Section 7.4.1.4. The cluster effect could not be avoided here due to the lack of use of higher-order methods.

The time required to obtain the solution is much higher (more than 100 times for $\varepsilon = 0.001$) than that required in Section 7.4.1.4 due to the necessity to perform numerical integration.

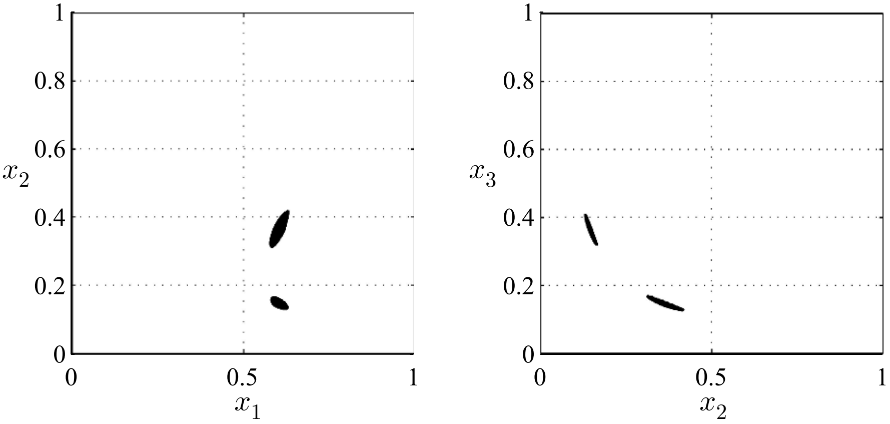


Fig. 7.2: Projection onto the (x_1,x_2) plane (left) and (x_2,x_3) plane (right) of a guaranteed outer approximation of the set of all global minimizers ($\varepsilon = 0.001$)

7.5 Conclusions and Perspectives

Interval analysis provides tools for the guaranteed characterization of the set of all global minimizers of the cost function associated with parameter estimation even when the model output is obtained via the numerical solution of a set of ordinary differential equations. This chapter has shown that when the cost function involves many occurrences of the parameters, as is usually the case in parameter estimation via optimization, higher-order techniques for box reduction and elimination, as well as reformulation of the cost function may play a very important role in reducing complexity.

Such tools are however still very difficult to employ when no explicit expression of the cost function is available. Their adaptation to models described by ODEs poses in principle no problem. It would for example be possible to use higher-order Taylor models [16]. Such models would be helpful to get closed-form enclosures of the cost function. Higher-order Taylor models have also better approximation properties than the methods described here. However, computing times are $O(n^k)$ in dimension n for order k. This usually takes too much time already in low dimensions, except if these models are used near the solution only. There, however, we use the exclusion box trick of Section 7.3.2 which is usually sufficient and can be performed with $O(n^3)$ effort. The combination of higher-order Taylor models with sensitivity functions, to get closed-form enclosures for gradients and Hessian matrices has also to be considered.

Acknowledgements This work was supported by the Austrian Science Found (FWF) Grants Nr. P18704-N13 and P22239-N13, by the Hungarian National Development Agency (NFÜ) Grant TÁMOP-4.2.2/08/1/2008-0008, by the ANR CPP, and by the Digiteo PASO project.

References

1. The COCONUT environment. URL www.mat.univie.ac.at/coconut-environment
2. Byrd, R., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput. **16**, 1190–1208 (1995)
3. Domes, F., Neumaier, A.: Rigorous enclosures of ellipsoids and directed cholesky factorizations (2009). URL http://www.mat.univie.ac.at/~dferi/publ/Cholesky.pdf. Manuscript
4. Domes, F., Neumaier, A.: Constraint propagation on quadratic constraints. Constraints **10**, 404–429 (2010)
5. Hansen, E.R.: Global Optimization Using Interval Analysis. Marcel Dekker, New York, NY (1992)
6. Hoefkens, J., Berz, M., Makino, K.: Efficient high-order methods for ODEs and DAEs. In: G. Corliss, C. Faure, A. Griewank (eds.) Automatic Differentiation : From Simulation to Optimization, pp. 341–351. Springer-Verlag, New-York, NY (2001)
7. Jacquez, J.A.: Compartmental Analysis in Biology and Medicine. Elsevier, Amsterdam (1972)
8. Kahan, W.: A more complete interval arithmetic. Lecture notes for a summer course, University of Toronto, Canada (1968)
9. Kay, S.M.: Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory. Prentice Hall (1993)
10. Kearfott, R.B.: INTLIB: a portable FORTRAN 77 elementary function library. ACM Transactions on Mathematical Software **20**(4), 447–459 (1994)
11. Kearfott, R.B.: Rigorous Global Search: Continuous Problems (Nonconvex Optimization and Its Applications. Springer-Verlag (2010)
12. Land, A.H., Doig, A.G.: An automated method for solving discrete programming problems. Econometrica **28**, 497–520 (1960)

13. Lin, Y., Stadtherr, M.A.: Validated solution of odes with parametric uncertainties. In: W. Marquardt, C. Pantelides (eds.) Proc. 16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering, *Computer Aided Chemical Engineering*, vol. 21, pp. 167 – 172. Elsevier (2006). DOI 10.1016/S1570-7946(06)80041-6. URL http://www.sciencedirect.com/science/article/B8G5G-4P37F2K-S/2/a22680956d2786784b23e88e9b272db6
14. Little, J.D., Murty, K.C., Sweeney, D.W., Karel, C.: An algorithm for the travelling salesman problem. Operations Research **11**, 972–989 (1963)
15. Lohner, R.: Computation of guaranteed enclosures for the solutions of ordinary initial and boundary value-problem. In: J.R. Cash, I. Gladwell (eds.) Computational Ordinary Differential Equations, pp. 425–435. Clarendon Press, Oxford (1992)
16. Makino, K., Berz, M.: Taylor models and other validated functional inclusion methods. International Journal **4**(4), 379–456 (2003)
17. Markót, M.C., Schichl, H.: Bound constrained optimization in the COCONUT environment (2010). Manuscript
18. Moore, R.E.: Interval arithmetic and automatic error analysis in digital computing. Phd thesis, Stanford University, Stanford, CA (1962). URL http://interval.louisiana.edu/Mooresearlypapers/disert.pdf
19. Müller, M.: Über das Fundamentaltheorem in der Theorie der gewöhnlichen Differentialgleichungen. Math. Z. **26**, 619–645 (1926)
20. Nedialkov, N.S., Jackson, K.R.: Methods for initial value problems for ordinary differential equations. In: U. Kulisch, R. Lohner, A. Facius (eds.) Perspectives on Enclosure Methods, pp. 219–264. Springer-Verlag, Vienna (2001)
21. Neumaier, A.: Interval Methods for Systems of Equations. Cambridge University Press, Cambridge, UK (1990)
22. Neumaier, A.: Complete search in continuous global optimization and constraint satisfaction. In: A. Iserles (ed.) Acta Numerica, pp. 271–369. Cambridge University Press (2004)
23. Ratschek, H., Rokne, J.: New Computer Methods for Global Optimization. Ellis Horwood, Chichester, UK (1988)
24. Ratz, D., Csendes, T.: On the selection of subdivision directions in interval branch-and-bound methods for global optimization. Journal of Global Optimization **7**(2), 183–207 (1995)
25. Rauh, A., Auer, E., Minisini, J., Hofer, E.P.: Extensions of ValEncIA-IVP for reduction of overestimation, for simulation of differential algebraic systems, and for dynamical optimization. Proc. Appl. Math. Mech. **7**(1), 1023,0011023,002 (2007). DOI 10.1002/pamm.200700022
26. Rauh, A., Hofer, E.P., Auer, E.: VALENCIA-IVP: A comparison with other initial value problem solvers. In: Proc. 12th GAMM - IMACS International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics (SCAN 2006), p. 36. IEEE Computer Society, Duisburg, Germany (2006). DOI 10.1109/SCAN.2006.47
27. Schichl, H.: Global optimization in the COCONUT project. In: Proceedings of the Dagstuhl Seminar "Numerical Software with Result Verification", *Lecture Notes in Comput. Sci.*, vol. 2991, pp. 277–293 (2004)
28. Schichl, H., Markót, M.C.: Algorithmic differentiation in the COCONUT environment (2010). URL http://www.mat.univie.ac.at/~herman/papers/griewank.pdf. Manuscript
29. Schichl, H., Markót, M.C.: Interval analysis on directed acyclic graphs for global optimization. higher order methods (2010). URL http://www.mat.univie.ac.at/~herman/papers/dag2.pdf. Manuscript
30. Schichl, H., Markót, M.C.: Optimal enclosures of derivatives and slopes for univariate functions (2010). URL http://www.mat.univie.ac.at/~herman/papers/univar.pdf. Manuscript
31. Schichl, H., Markót, M.C., Neumaier, A.: Exclusion regions for optimization problems (2010). URL http://www.mat.univie.ac.at/~herman/papers/exclopt.pdf. Manuscript

32. Schichl, H., Neumaier, A.: Exclusion regions for systems of equations. SIAM J. Numer. Anal. **42**, 383–408 (2004)
33. Schichl, H., Neumaier, A.: Interval analysis on directed acyclic graphs for global optimization. J. Global Optim. **33**, 541–562 (2006)
34. Shen, Z., Neumaier, A.: The Krawczyk operator and Kantorovich's theorem. J. Math. Anal. Appl. **149**, 437–443 (1990)
35. Skelboe, S.: Computation of rational interval functions. BIT **14**, 87–95 (1974)
36. Vu, X.H., Schichl, H., Sam-Haroud, D.: Using directed acyclic graphs to coordinate propagation and search for numerical constraint satisfaction problems. In: Proc. 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004). Florida (2004)
37. Walter, E., Kieffer, M.: Guaranteed optimisation of the parameters of continuous-time knowledge-based models. In: C. Commault, N. Marchand (eds.) Positive Systems, *LNCIS*, vol. 341, pp. 137–144. Springer, Heidelberg (2006)
38. Walter, E., Pronzato, L.: Identification of Parametric Models from Experimental Data. Springer-Verlag, London (1997)
39. Wolfe, M.A.: Interval methods for global optimization. Applied Mathematics and Computation **75**, 179–206 (1996)

Chapter 8
Optimal Control of Induction Heating: Theory and Application

Darya Filatova (✉) and Marek Grzywaczewski

Abstract The theoretic background of an optimal control task for a precision induction heating problem is studied in this work. The basics of electro-magnetic and heat transfer theory are used to describe the dynamics of induction heating processes of rectangle workpieces. The main result of this work, presented as the first-order necessary conditions for the optimal solution of the considered control task, allows one to employ interval representations of the mathematical model's main parameters in order to study the influence of environment uncertainties which have dominant effects on induction heating processes.

8.1 Introduction

The state-of-the-art of manufacturing technologies requires control techniques which improve product quality with reduced energy consumption, maximize productivity under consideration of environmental constraints, and optimize consistency by extending the fixture life cycle. Induction heating is one of the progressive processes which are applicable to bond, harden or soften metals or other conductive materials.

It is clear that the mathematical model of the induction heating has to take into account electromagnetic, thermal and metallurgical phenomena and, thus usually consists of two equations, namely a first one is for the inductor electromagnetic field and a second one for thermal phenomenon in a heated part. Both equations are related by means of the electrical resistivity parameter. The values of the parameters

Darya Filatova
UJK, ul. Krakowska 11, 25-027 Kielce, Poland and Analytical Centre of Russian Academy of Sciences, ul. Vavilova 40, 199911 Moscow, Russia
e-mail: daria_filatova@rambler.ru

Marek Grzywaczewski
Politechnika Radomska, ul. Malczewskiego 20A, 26-600 Radom, Poland
e-mail: mgrzyw@interia.pl

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_8,

very often depend on the temperature of the heated part, the purity of its material, and the environmental temperature. In addition, heat losses from conduction, convection and radiation should also be considered, for an example, by means of an interval representation of the technological parameters. All this causes complications in the solution of induction heating tasks especially if they are to be solved in an optimal manner. An optimal control task in the general form can be formulated as follows.

Let $\mathbf{x} : [t_0, t_1] \to \mathbb{R}^n$ be a state vector related to space coordinates of a heated workpiece and defined on a time interval $I = [t_0, t_1]$, a heating temperature function $\mathbb{T}(\mathbf{x}, t) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^{n+1}$ which is twice differentiable by $\mathbf{x}$ and once differentiable by $t \in I$. The trajectory of $\mathbb{T}(\mathbf{x}, t)$ is subject to a control vector $\mathbf{u} = (\mathbf{u}_1(\mathbf{x}, t), \mathbf{u}_2(t))$, where $\mathbf{u}_1(\mathbf{x}, t) \in \mathbb{R}^r$ presents the control of the different sections of the induction coil and $\mathbf{u}_2(t) \in \mathbb{R}^s$ presents the control of the whole capacity of the induction coil. The production management wants to choose $\mathbf{u}$ in such a way that the quality of the production is optimized over the planning horizon I. Namely, the following functional is to be minimized

$$J[\mathbb{T}(\cdot,\cdot), \mathbf{u}(\cdot)] \to \min_{\mathbf{u}}, \tag{8.1}$$

where $J[\cdot,\cdot]$ is defined on the domain $\Upsilon = \Omega \times I$, where $\Omega \subset \mathbb{R}^n$ and $I \in \mathbb{R}$ take into account the following conditions.

a) The dynamics of a system under consideration or the object equation is given by

$$\mathscr{L}[\mathbb{T}] = \mathbb{S}(\mathbf{u}, \mathbf{x}, t, \mathbb{T}), \tag{8.2}$$

where $\mathscr{L}[\cdot]$ is a differential operator of parabolic type, $\mathbb{S}(\cdot,\cdot,\cdot,\cdot)$ is an elliptic operator and $(\mathbf{x}, t) \in \Upsilon$.

b) The set of all possible initial and terminal values of the trajectory $\mathbb{T}$ is defined by

$$K_1(\mathbb{T}(\mathbf{x}, t_0), \mathbb{T}(\mathbf{x}, t_1)) \leq 0, \tag{8.3}$$

$$K_2(\mathbb{T}(\mathbf{x}, t_0), \mathbb{T}(\mathbf{x}, t_1)) = 0, \tag{8.4}$$

where $\mathbf{x} \in \Omega$, $K_1 : \mathbb{R}^{n+1} \times \mathbb{R}^{n+1} \to \mathbb{R}^{k_1}$ and $K_2 : \mathbb{R}^{n+1} \times \mathbb{R}^{n+1} \to \mathbb{R}^{k_2}$ are some functions.

c) The boundary conditions are presented by

$$\ell[\mathbb{T}] = q(\mathbf{u}, \mathbf{x}, t, \mathbb{T}), \tag{8.5}$$

where $\ell(\cdot)$ is a differential operator and $q(\cdot,\cdot,\cdot,\cdot)$ is a function which corresponds to physical phenomenon conditions; $\mathbf{x} \in \Gamma$ ($\Gamma = \partial\Omega$ is a piecewise smooth bound), $t \in I$.

d) The control constraints for $i = 1, 2, ..., k$ and for any $(\mathbf{x}, t) \in \Upsilon$ are given

$$\varphi_i(\mathbf{u}(\mathbf{x}, t), \mathbf{x}, t) \leq 0, \tag{8.6}$$

where $(\mathbf{x},t) \in \Upsilon$.

e) The terminal constraints are formulated as

$$\kappa(\mathbb{T}(\mathbf{x},t_1),\mathbb{T}^*(\mathbf{x})) \leq 0, \tag{8.7}$$

where $\mathbb{T}^*(\mathbf{x})$ is the final temperature required from the technological point of view given a priori , $\mathbf{x} \in \Omega$.

f) The phase constraints take a form

$$\Phi_j(\mathbb{T}(\mathbf{x},t),\mathbf{x},t) \leq 0, \tag{8.8}$$

where $j = 1,2,...,m$, $(\mathbf{x},t) \in \Upsilon$.

A good review on the considered optimal control problem and its chronological development can be found in [13]. Here, we comment only on several aspects connected to the induction heating problem. So, first attempts to solve the problem by means of the maximum principle belong to Butkovskii's group [1,2]. However, the results were far from technological process requirements. The next step in the problem solution, done by Rapoport's group, allowed to get an improved solution by means of the alternance method [14], [16]. Both methods deal with a simple geometry of the shapes and constant parameters of heated part. Moreover, the theoretical backgrounds for the problem solution, where the control action enters to the system dynamics through the boundary conditions, have not been shown. Recent developments in optimal control theory allow one to get the first-order necessary optimality conditions for the task (8.1) - (8.8) using the Dubovitski-Milyutin method. Our main task in this research is to obtain these conditions of optimality and to show how to adapt the theory for cases when the materials exhibit a strong dependence on the temperature [15].

The rest of the paper is organized as follows. In Section 2, the precision heating problem is formulated with respect to the heat transmitting equation and the predetermined heating sources. Section 3 presents the first-order necessary conditions for the optimal solution based on the Dubovitski-Milyutin method taking into account constraints (8.3) - (8.8). The parametrization of the control, its transformation to a bang-bang control problem and a short illustrative example are shown in Sections 4 and 5. Finally, some conclusions are presented in Section 6.

8.2 Precision Induction Heating Problem

Let us reformulate the task of induction heating concerning only the case of precision heating and taking into account technological process constraints. In this case, the goal function (8.1) takes the form

$$J(\mathbb{T}(\mathbf{x},t_1),\mathbf{u}) = \max_{\Omega} |\mathbb{T}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x})| \to \min_{u} \tag{8.9}$$

and is subject to the following constraints, namely

a) the model (8.2) of the object takes the form

$$c\left(\mathbb{T}\left(\mathbf{x},t\right)\right)\frac{\partial\mathbb{T}\left(\mathbf{x},t\right)}{\partial t}-\nabla\lambda\left(\mathbb{T}\left(\mathbf{x},t\right)\right)\nabla\mathbb{T}\left(\mathbf{x},t\right)=\phi\left(\mathbf{u}\left(\mathbf{x},t\right),\mathbf{x},t,\mathbb{T}\right) \tag{8.10}$$

with initial conditions $\mathbb{T}\left(\mathbf{x},t_0\right)=\mathbb{T}_0\left(\mathbf{x}\right)$ for any $\mathbf{x}\in\Omega$;

b) the constraints by the cost of the industrial production for $t=t_1$

$$\int\limits_{\Upsilon}\Phi_0\left(\mathbf{u}\left(\mathbf{x},t\right),\mathbf{x},t,\mathbb{T}\right)d\mathbf{x}dt\leq\xi_1, \tag{8.11}$$

where $\xi_1\geq 0$;

c) the boundary conditions are

$$\lambda\left(\mathbb{T}\left(\mathbf{x},t\right)\right)\frac{\partial\mathbb{T}}{\partial n}=q\left(\mathbf{u}\left(\mathbf{x},t\right),\mathbf{x},t,\mathbb{T}\right) \tag{8.12}$$

for any $\mathbf{x}\in\Gamma$ and $t\in I$ ($\frac{\partial(\cdot)}{\partial n}$ is an operator of the directional derivative of a function);

d) the control constraints for any $(\mathbf{x},t)\in\Upsilon$ take the form

$$\varphi_i\left(\mathbf{u}\left(\mathbf{x},t\right),\mathbf{x},t,\mathbb{T}\right)\leq 0 \tag{8.13}$$

for $i=1,2,...,k$;

e) the terminal constraints for $t=t_1$ are given as

$$\max_{\mathbf{x},\mathbf{y}}\left|\mathbb{T}\left(\mathbf{x},t\right)-\mathbb{T}\left(\mathbf{y},t\right)\right|\leq\varepsilon \tag{8.14}$$

for any $(\mathbf{x},t)\in\Upsilon\cap(\mathbf{y},t)\in\Upsilon$, $\varepsilon>0$;

f) the phase constraints are

$$\max_{\Upsilon}\mathbb{T}\left(\mathbf{x},t\right)\leq\xi_2 \tag{8.15}$$

for any $(\mathbf{x},t)\in\Upsilon$ and $\xi_2\geq 0$.

8.3 The Local Maximum Principle

As already described, we are going to use the Dubovitski-Milyutin method [10], [11] to solve the problem (8.9) - (8.13). This means that first of all we have to get the Euler equation, after the Pontryagin function, and at last to formulate the first-order necessary optimality conditions.

8.3.1 Preliminaries

Define the set

$$V[t_0,t_1] := \{\mathbf{u} : [t_0,t_1] \to \mathbb{U} | \mathbf{u}(\cdot) \text{ is measurable}\},$$

where $\mathbb{U}$ can be regarded as a metric space. Any $\mathbf{u}(\cdot) \in V[t_0,t_1]$ is called a feasible control.

Assumption 1. A control $\mathbf{u}(\cdot)$ is called an admissible control, and $(\mathbb{T},\mathbf{u})$ is called an admissible pair, if:

- $\mathbf{u}(\cdot) \in V[t_0,t_1]$,
- the object equation (8.10) has a unique solution under $\mathbf{u}(\cdot)$,
- the constraints (8.10) - (8.13) are satisfied, and
- the functional (8.9) belongs to the set of Lebesgue measurable functions ρ : $[t_0,t_1] \to \mathbb{R}^n$ such that $\int\limits_{t_0}^{t_1} |\rho(t)|^p dt < \infty$ $(p \in [1,\infty))$. ■

The set of all admissible controls is denoted by $V_{ad}[t_0,t_1]$. Now, the task of optimal control can be formulated as follows:

$$\textbf{minimize (8.9) over } V[t_0,t_1].$$

Assumption 2. The problem is said to be finite if (8.9) has a finite lower bound, and it is said to be solvable if there is a $\mathbf{u}^{opt}(\cdot) \in V_{ad}[t_0,t_1]$ satisfying

$$J(\mathbf{u}^*(\cdot)) = \inf_{\mathbf{u}(\cdot)\in V[t_0,t_1]} J(\mathbf{u}(\cdot)). \tag{8.16}$$

Any $\mathbf{u}^{opt}(\cdot) \in V[t_0,t_1]$ satisfying (8.16) is called an optimal control and $[\mathbb{T}^{opt}(\cdot,\cdot),$ $\mathbf{u}^{opt}(\cdot)]$ is called an optimal pair. ■

8.3.2 The Euler Equation Analysis

Suppose that all conditions of **Assumption 1** and **Assumption 2** are fulfilled, moreover, γ is some integral function, $\psi(\mathbf{x},t) : \Upsilon \to \mathbb{R}^1$ is a function, dv, $d\theta$ and $d\mu$ are non-negative Radon measures on $\Upsilon \subset \mathbb{R}^{n+1}$, α_0 is a Lagrange multiplier, $m_i(\mathbf{x},t)$ is a non-negative function which takes zero values only on the set

$$M(\varphi_i) = \{(\mathbf{x},\mathbf{t}) \in \Upsilon : \varphi_i(\mathbf{u}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}) = 0\}),$$

for $i = 1,2,...,k$. Then, taking into account the object model (8.10) and the constraints (8.12) and (8.13), the Euler equation takes the form[1]

[1] The index var means ”variation” throughout this paper to distinguish it from the notation of the interval estimate. For the simplicity of the reasoning, we omit the arguments of the functions and only state them if they are required to emphasize the mathematical meaning.

$$
\begin{aligned}
&\int\limits_{\Upsilon}\left(\left(\lambda\nabla^2\psi+c\frac{\partial\psi}{\partial t}\right)\mathbb{T}^{var}+\psi\phi_{\mathbf{u}}'\mathbf{u}^{var}+\psi\phi_{\mathbb{T}}'\mathbb{T}^{var}\right)d\mathbf{x}dt\\
&+\int\limits_{\Gamma\times I}\left(\lambda\frac{\partial\psi}{\partial n}-\psi\frac{\partial\lambda}{\partial n}\right)\mathbb{T}^{var}d\Gamma dt\\
&-\int\limits_{\Gamma\times I}\psi\lambda\frac{\partial\mathbb{T}^{var}}{\partial n}d\Gamma dt-\int\limits_{\Omega}c\psi\mathbb{T}^{var}\Big|_{t_0}^{t_1}d\mathbf{x}\\
&+\gamma\left(\frac{\partial\lambda}{\partial n}\mathbb{T}^{var}+\lambda\frac{\partial\mathbb{T}^{var}}{\partial n}-q_{\mathbb{T}}'\mathbb{T}^{var}\right)+\beta\left(\mathbb{T}^{var}\left(\mathbf{x},t_0\right)\right)\\
&-\int\limits_{\Upsilon}\mathbb{T}^{var}d\mu-\sum_i\left\langle m_i,\varphi_{i\mathbf{u}}'\mathbf{u}^{var}\right\rangle-\int\limits_{\Omega}\mathbb{T}^{var}\left(\mathbf{x},t_1\right)dv\\
&-\int\limits_{\Upsilon}\mathbb{T}^{var}d\theta-\alpha_0\int\limits_{\Upsilon}\left(\Phi_{o\mathbf{u}}'\mathbf{u}^{var}+\Phi_{o\mathbb{T}}'\mathbb{T}^{var}\right)d\mathbf{x}dt=0,
\end{aligned}
\tag{8.17}
$$

where $\beta\left(\mathbb{T}^{var}\left(\mathbf{x},t_0\right)\right)$ and γ are some linear functionals.

Let $\mathbb{T}^{var}\left(\mathbf{x},t\right)=0$ and $\mathbf{u}^{var}$ be arbitrary for any $\left(\mathbf{x},t\right)\in\Upsilon$. In this case (8.17) takes form

$$
\int\limits_{\Upsilon}\psi\phi_{\mathbf{u}}'\mathbf{u}^{var}d\mathbf{x}dt-\sum_i\left\langle m_i,\varphi_{i\mathbf{u}}'\mathbf{u}^{var}\right\rangle-\alpha_0\int\limits_{\Upsilon}\left(\Phi_{0i\mathbf{u}}'\mathbf{u}^{var}+\Phi_{0i\mathbb{T}}'\mathbb{T}^{var}\right)d\mathbf{x}dt=0.
$$

According to the Yosida-Hewitt theorem [8], any functional m_i can be presented as

$$
m_i=m_i^a+m_i^s,
$$

where m_i^a is an absolutely continuous component and m_i^s is a singular component of the functional m_i, $i=1,2,...,k$.

Let the function $m_i\left(\mathbf{x},t\right):\Upsilon\longrightarrow\mathbb{R}^1$, $i=1,2,...,k$, fulfill the condition

$$
\left\langle m_i^a,\xi\right\rangle=\int\limits_{\Upsilon}m_i\left(\mathbf{x},t\right)\xi\left(\mathbf{x},t\right)d\mathbf{x}dt
$$

for any $\xi\left(\cdot,\cdot\right)\in L^{\infty}\left(\Upsilon\right)$. So $m_i\left(\mathbf{x},t\right)$ is a non-negative function, which takes zero values only on the set $M\left(\varphi_i\right)$, therefore the condition of complementary slackness is

$$
m_i\left(\mathbf{x},t\right)\varphi_i\left(\mathbf{u}\left(\mathbf{x},t\right),\mathbf{x},t\right)=0.
$$

This yields

$$
\int\limits_{\Upsilon}\left(\psi\phi_{\mathbf{u}}'-\alpha_0\Phi_{o\mathbf{u}}'\right)\mathbf{u}^{var}d\mathbf{x}dt-\int\limits_{\Upsilon}\sum_i m_i\varphi_{\mathbf{u}}'\mathbf{u}^{var}d\mathbf{x}dt=0.
$$

Recall that $\mathbf{u}=\left(\mathbf{u}_1,\mathbf{u}_2\right)$, so

$$
\begin{aligned}
0=&\int\limits_{\Upsilon}\left(\psi\phi_{\mathbf{u}_1}'-\alpha_0\Phi_{o\mathbf{u}_1}'-\sum_i m_i\varphi_{\mathbf{u}_1}'\right)\mathbf{u}_1^{var}d\mathbf{x}dt\\
&-\int\limits_{\Upsilon}\left(\psi\phi_{\mathbf{u}_2}'-\alpha_0\Phi_{o\mathbf{u}_2}'-\sum_i m_i\varphi_{\mathbf{u}_2}'\right)\mathbf{u}_2^{var}d\mathbf{x}dt.
\end{aligned}
$$

Taking $\mathbf{u}_2^{var} = 0$, we have

$$\int\limits_{\Upsilon} \left(\psi \phi'_{\mathbf{u}_1} - \alpha_0 \Phi'_{o\mathbf{u}_1} - \sum_i m_i \varphi'_{\mathbf{u}_1} \right) \mathbf{u}_1^{var} d\mathbf{x} dt = 0$$

so

$$\psi \phi'_{u_1} - \alpha_0 \Phi'_{ou_1} - \sum_i m_i \varphi'_{u_1} = 0.$$

If $\mathbf{u}_1^{var} = 0$ holds, we have

$$\int\limits_{I} \left[\int\limits_{\Omega} \left(\psi \phi'_{\mathbf{u}_2} - \alpha_0 \Phi'_{o\mathbf{u}_2} - \sum_i m_i \varphi'_{\mathbf{u}_2} \right) d\mathbf{x} \right] \mathbf{u}_2^{var} dt,$$

which gives

$$\int\limits_{\Omega} \left(\psi \phi'_{\mathbf{u}_2} - \alpha_0 \Phi'_{o\mathbf{u}_2} - \sum_i m_i \varphi'_{\mathbf{u}_2} \right) d\mathbf{x} = \mathbf{0}.$$

Now we introduce the Pontryagin functions

$$H = \psi \phi'_{\mathbf{u}} - \alpha_0 \Phi'_{o\mathbf{u}} - \sum_i m_i \varphi'_{\mathbf{u}}$$

and

$$h = \int\limits_{\Omega} H d\mathbf{x}.$$

This leads to the local maximum principle for the component $\mathbf{u}_1$

$$H'_{\mathbf{u}_1} = 0$$

as well as for the component $\mathbf{u}_2$

$$h'_{\mathbf{u}_2} = 0.$$

After setting $\mathbf{u}^{var} = 0$ the Euler equation (8.17) can be rewritten according to

$$\begin{aligned}
&\int\limits_{\Upsilon}\left(\left(\lambda\nabla^2\psi+c\tfrac{\partial\psi}{\partial t}\right)\mathbb{T}^{var}+\psi\phi_{\mathbb{T}}'\mathbb{T}^{var}\right)d\mathbf{x}dt\\
&+\int\limits_{\Gamma\times I}\left(\lambda\tfrac{\partial\psi}{\partial n}-\psi\tfrac{\partial\lambda}{\partial n}\right)\mathbb{T}^{var}d\Gamma dt-\\
&-\int\limits_{\Gamma\times I}\psi\lambda\tfrac{\partial\mathbb{T}^{var}}{\partial n}d\Gamma dt-\int\limits_{\Omega}c\psi\mathbb{T}^{var}\Big|_{t_0}^{t_1}d\mathbf{x}\\
&+\gamma\left(\tfrac{\partial\lambda}{\partial n}\mathbb{T}^{var}+\lambda\tfrac{\partial\mathbb{T}^{var}}{\partial n}-q_{\mathbb{T}}'\mathbb{T}^{var}\right)+\beta\left(\mathbb{T}^{var}\left(\mathbf{x},t_0\right)\right)\\
&-\int\limits_{\Upsilon}\mathbb{T}^{var}d\mu-\int\limits_{\Omega}\mathbb{T}^{var}\left(\mathbf{x},t_1\right)d\nu-\int\limits_{\Upsilon}\mathbb{T}^{var}d\theta\\
&-\alpha_0\int\limits_{\Upsilon}\left(\Phi_{o\mathbf{u}}'\mathbf{u}^{var}+\Phi_{o\mathbb{T}}'\mathbb{T}^{var}\right)d\mathbf{x}dt=0.
\end{aligned}\tag{8.18}$$

To get the adjoint equation, we set $\mathbb{T}^{var}\left(\mathbf{x},t\right)=0$ and $\frac{\partial\mathbb{T}^{var}(\mathbf{x},t)}{\partial n}=0$ in (8.18) for any $\left(\mathbf{x},t\right)\in\Gamma\times I$ and allow arbitrary values $\mathbb{T}^{var}\left(\mathbf{x},t\right)$ for any $\left(\mathbf{x},t\right)\in\Upsilon\backslash\Gamma\times I$, so that

$$\int\limits_{\Upsilon}\left(\lambda\nabla^2\psi+c\frac{\partial\psi}{\partial t}+\psi\phi_{\mathbb{T}}'\right)\mathbb{T}^{var}d\mathbf{x}dt-\int\limits_{\Upsilon}\mathbb{T}^{var}d\mu-\int\limits_{\Upsilon}\mathbb{T}^{var}d\theta=0\tag{8.19}$$

results. Rewriting the equation (8.19), we have

$$\left(\lambda\nabla^2\psi+c\frac{\partial\psi}{\partial t}+\psi\phi_{\mathbb{T}}'\right)d\mathbf{x}dt=d\mu+d\theta.$$

Now setting $\mathbf{u}^{var}=0$ in (8.18), we come to

$$\begin{aligned}
&\int\limits_{\Gamma\times I}\left(\lambda\tfrac{\partial\psi}{\partial n}-\psi\tfrac{\partial\lambda}{\partial n}\right)\mathbb{T}^{var}d\Gamma dt-\int\limits_{\Gamma\times I}\psi\lambda\tfrac{\partial\mathbb{T}^{var}}{\partial n}d\Gamma dt-\int\limits_{\Omega}c\psi\mathbb{T}^{var}\Big|_{t_0}^{t_1}d\mathbf{x}\\
&+\gamma\left(\tfrac{\partial\lambda}{\partial n}\mathbb{T}^{var}+\lambda\tfrac{\partial\mathbb{T}^{var}}{\partial n}-q_{\mathbb{T}}'\mathbb{T}^{var}\right)+\beta\left(\mathbb{T}^{var}\left(\mathbf{x},t_0\right)\right)-\int\limits_{\Omega}\mathbb{T}^{var}\left(\mathbf{x},t_1\right)d\nu=0,
\end{aligned}\tag{8.20}$$

where $\gamma(\cdot)$ is some integral functional, such that

$$\gamma(z)=\int\limits_{\Gamma\times I}\widetilde{\gamma}zd\Gamma dt$$

and, therefore,

$$\int\limits_{\Gamma\times I}\psi\lambda\frac{\partial\mathbb{T}^{var}}{\partial n}d\Gamma dt=\int\limits_{\Gamma\times I}\widetilde{\gamma}zd\Gamma dt.$$

In addition, the condition $\mathbb{T}^{var}\left(\mathbf{x},t\right)=0$ holds for any $\left(\mathbf{x},t\right)\in\Gamma\times I$; $\mathbb{T}^{var}\left(\mathbf{x},t_0\right)=0$ as well as $\mathbb{T}^{var}\left(\mathbf{x},t_1\right)$ for any $\left(\mathbf{x},t\right)\in\Omega$; since $\frac{\partial\overline{\mathbb{T}}(\mathbf{x},t)}{\partial n}$ is arbitrary for any $\left(\mathbf{x},t\right)\in\Gamma\times I$, we have the possibility to rewrite (8.20) as

$$-\int_{\Gamma\times I} \psi\lambda \frac{\partial \mathbb{T}^{var}}{\partial n} d\Gamma dt + \gamma\left(\frac{\partial \lambda}{\partial n}\mathbb{T}^{var}\right) = 0. \tag{8.21}$$

Taking into account that $\lambda(\mathbb{T}) > 0$ is satisfied on $\Gamma \times I$, we get $\psi = \widetilde{\gamma}$ for any $(\mathbf{x},t) \in \Gamma \times I$ and, therefore,

$$\gamma(z) = \int_{\Gamma\times I} \psi z d\Gamma dt. \tag{8.22}$$

Now rewrite (8.20) by using (8.21) and (8.22)

$$\begin{aligned}&\int_{\Gamma\times I}\left(\lambda\tfrac{\partial\psi}{\partial n} - \psi\tfrac{\partial\lambda}{\partial n}\right)\mathbb{T}^{var} d\Gamma dt - \int_{\Omega} c\psi\mathbb{T}^{var}\,\Big|_{t_0}^{t_1} d\mathbf{x}\\ &+ \int_{\Gamma\times I}\psi\left(\tfrac{\partial\lambda}{\partial n} - q'_{\mathbb{T}}\right)\mathbb{T}^{var} d\Gamma dt + \beta\left(\mathbb{T}^{var}(\mathbf{x},t_0)\right) - \int_{\Omega}\mathbb{T}^{var}(\mathbf{x},t_1)\,dv = 0.\end{aligned}$$

Setting $\mathbb{T}^{var}(\mathbf{x},t_0) = 0$ and $\mathbb{T}^{var}(\mathbf{x},t_1)$ for any $(\mathbf{x},t) \in \Omega$ gives

$$\int_{\Gamma\times I}\left(\lambda\frac{\partial\psi}{\partial n} - \psi\frac{\partial\lambda}{\partial n}\right)\mathbb{T}^{var} d\Gamma dt + \int_{\Gamma\times I}\psi\left(\frac{\partial\lambda}{\partial n} - q'_{\mathbb{T}}\right)\mathbb{T}^{var} d\Gamma dt = 0,$$

so

- for any $\mathbb{T}^{var}$

$$\int_{\Gamma\times I}\left(\lambda\frac{\partial\psi}{\partial n} - \psi q'_{\mathbb{T}}\right)\mathbb{T}^{var} d\Gamma dt = 0,$$

- for any $(\mathbf{x},t) \in \Gamma \times I$

$$\lambda\frac{\partial\psi}{\partial n} - \psi q'_{\mathbb{T}} = 0.$$

Additionally the Euler equation takes new form

$$-\int_{\Omega} c\psi\mathbb{T}^{var}\,\Big|_{t_0}^{t_1} d\mathbf{x} + \beta\left(\mathbb{T}^{var}(\mathbf{x},t_0)\right) - \int_{\Omega}\mathbb{T}^{var}(\mathbf{x},t_1)\,dv = 0. \tag{8.23}$$

Let $\mathbb{T}^{var}(\mathbf{x},t_0)$ take any values for any $\mathbf{x} \in \Omega$ and $\mathbb{T}^{var}(\mathbf{x},t_1) = 0$ for any $\mathbf{x} \in \Omega$. The equation (8.23) can be rewritten as

$$-\int_{\Omega} c(\mathbb{T})\,\psi(\mathbf{x},t_0)\,\mathbb{T}^{var}(\mathbf{x},t_0)\,d\mathbf{x} + \beta\left(\mathbb{T}^{var}(\mathbf{x},t_0)\right) = 0. \tag{8.24}$$

This means that $c(\mathbb{T})\,\psi(\mathbf{x},t_0)$ can take any values. To complete the analysis we show how to get the transversality conditions using the rest of the equation (8.24) given as

$$-\int_{\Omega} c(\mathbb{T})\,\psi(\mathbf{x},t_1)\,\mathbb{T}^{var}(\mathbf{x},t_1)\,d\mathbf{x} - \int_{\Omega} \mathbb{T}^{var}(\mathbf{x},t_1)\,d\nu = \mathbf{0}$$

for any $\mathbb{T}^{var}(\mathbf{x},t_1)$.

Suppose that $\mathbb{T}^{var}(\mathbf{x},t_1)$ is arbitrary for any $\mathbf{x}\in\Omega$. This gives

$$-c(\mathbb{T})\,\psi(\mathbf{x},t_1)\,d\mathbf{x} - d\nu = 0$$

or equivalently

$$\psi(\mathbf{x},t_1)\,d\mathbf{x} = -\frac{d\nu}{c(\mathbb{T})}.$$

Let $d\nu_1 = -\frac{d\nu}{c(\mathbb{T})}$. This measure has the properties $d\nu_1 \geq 0$ and $d\nu_1(|\mathbb{T}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x})| - \varepsilon) = 0$, where $\varepsilon > 0$. The complementarity slackness is

$$\begin{array}{lll} d\nu_1 = 0 & \text{if} & |\mathbb{T}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x})| - \varepsilon \neq 0, \\ d\nu_1 \neq 0 & \text{if} & \begin{cases} \mathbb{T}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x}) - \varepsilon = 0, \\ -(\mathbb{T}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x})) - \varepsilon = 0. \end{cases} \end{array}$$

Again, according to the Yosida-Hewitt decomposition theorem, we set

$$d\nu_1 = d\nu_1^a + d\nu_1^s,$$

where in our case $d\nu_1^a = 0$ and $d\nu_1^s \neq 0$ hold.

So that

$$\psi(\mathbf{x},t_1) = -\sum_{i=1}^{r} \frac{\mathbb{T}(\mathbf{x}_i,t_1) - \mathbb{T}^*(\mathbf{x}_i)}{\varepsilon}\,\delta(\mathbf{x}_i), \tag{8.25}$$

where $\delta(\cdot)$ is the Dirac function, $i = 1,2,...,k$ is an index of the active constraint.

The main result of this work can be formulated as follows.

Theorem 8.1. *Let* $(\mathbb{T}^{opt},\mathbf{u}^{opt})$ *be an optimal pair and let the property* $\lambda(\mathbb{T}^{opt}) \neq 0$ *holds on the envelope* Υ *and* $c(\mathbb{T}^{opt})$ *on* Υ*, then there exists a non-trivial set of Lagrange multipliers* α_0*,* $\psi(\mathbf{x},t)$*,* $d\nu$*,* $d\mu$*,* $m_i(\mathbf{x},t_1)$*,* $i = 1,2,...,k$*, such that the number* $\alpha_0 \geq 0$ *satisfies the complementarity slackness*

$$\alpha_0\left(\int_{\Upsilon} \Phi_0(u(x,t),x,t,\mathbb{T})\,dxdt - \xi_1\right) = 0,$$

$\psi(\mathbf{x},t) : \Upsilon \to \mathbb{R}^1$ is a function, dv is a non-negative Radon measure on $\Upsilon \subset \mathbb{R}^{n+1}$ concentrated on the set $M_{\mathbb{T}} = \{\mathbf{x} \in \Omega : |\mathbb{T}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x})| = \varepsilon\}$, $d\mu \geq 0$ is a nonnegative Radon measure on $\Upsilon \subset \mathbb{R}^{n+1}$ concentrated on the set

$$(\mathbf{x},t) \in \Upsilon \left| \mathbb{T}^{opt}(\mathbf{x},t) = \xi_2, \right.$$

$m_i(\mathbf{x},t_1) \to \mathbb{R}^1$ are nonnegative functions which satisfy complementarity slackness $m_i(\mathbf{x},t_1)\,\varphi_i(\mathbf{u}^{opt}(\mathbf{x},t),t) = 0$, $i = 1,2,...,k$, and the following conditions are fulfilled:

a) The adjoint equation is given by

$$\left(\lambda\left(\mathbb{T}^{opt}\right)\nabla^2\psi(\mathbf{x},t) + c\left(\mathbb{T}^{opt}\right)\frac{\partial\psi(\mathbf{x},t)}{\partial t} + \psi(\mathbf{x},t)\,\phi'_{\mathbb{T}} - \alpha\Phi'_{0\mathbb{T}}\right)d\mathbf{x}dt = d\mu. \tag{8.26}$$

b) The transversality condition, that is the boundary condition for $t = t_1$, is equal to

$$\psi(\mathbf{x},t_1)\,d\mathbf{x} = -\left(\mathbb{T}^{opt}(\mathbf{x},t_1) - \mathbb{T}^*(\mathbf{x})\right)dv \tag{8.27}$$

for any $x \in \Omega$.

c) The boundary conditions on $\Gamma \times I$ can be formulated by

$$\frac{\partial\psi}{\partial n} = \psi\, q'_{\mathbb{T}} \tag{8.28}$$

for any $(\mathbf{x},t) \in \Gamma \times I$.

d) Finally, the local maximum principle takes one of the following two forms, namely, either

$$\begin{aligned}
&\psi(\mathbf{x},t)\left(\phi'_{\mathbf{u}_2}(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T}) - \widetilde{q}'_{\mathbf{u}_2}(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T})\right)\\
&-\alpha_0\Phi'_{o\mathbf{u}_2}(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T})\\
&-\sum_{i=1}^{k} m_i(\mathbf{x},t)\,\varphi'_{i\mathbf{u}_2}(\mathbf{u}^{opt}(x,t),\mathbf{x},t) = 0;
\end{aligned} \tag{8.29}$$

for $(\mathbf{x},t) \in \Upsilon$, or

$$\begin{aligned}
&\int_{\Omega}\left[\psi(\mathbf{x},t)\left(\phi'_{\mathbf{u}_2}(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T}) - \widetilde{q}'_{\mathbf{u}_2}(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T})\right) -\right.\\
&\left.-\alpha_0\Phi'_{o\mathbf{u}_1}(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T})\right]dx\\
&-\int_{\Omega}\sum_{i=1}^{k} m_i(\mathbf{x},t)\,\varphi'_{i\mathbf{u}_1}(\mathbf{u}^{opt}(x,t),\mathbf{x},t) = 0,
\end{aligned} \tag{8.30}$$

with

$$\widetilde{q}(\mathbf{u},t,\mathbf{x},t,\mathbb{T}) = \begin{cases} q(\mathbf{u},t,\mathbf{x},t,\mathbb{T}) & if \quad \mathbf{x} \in \Gamma, \\ 0 & if \quad \mathbf{x} \in \Omega \backslash \Gamma. \end{cases} \tag{8.31}$$

for $t \in I$. ■

8.4 The Bang-Bang Control Case

The numerical solution of the induction heating task, even in less complicated cases has many obstacles [4] – [6], [9]. For the task (8.10), (8.26) – (8.31) the main result cannot be directly applied, especially in the case when the parameters of the induction heating process are not exactly defined. To adapt the theoretical solution to the situation mentioned above we convert the induction heating control problem to the bang-bang case. This requires linearity of the control and a specially selected protocol for the inductor, which can be presented by an interval.

Let the Pontryagin function have the form

$$H(\alpha_0, \psi(\mathbf{x},t), \mathbf{u}, \mathbf{x}, t) = \psi\phi(\mathbf{u},\mathbf{x},t) - \alpha_0 \Phi_0(\mathbf{u},\mathbf{x},t). \tag{8.32}$$

The functions $H\left(\alpha_0, \psi(\mathbf{x},t), \mathbf{u}_1, \mathbf{u}_2^{opt}(t), \mathbf{x}, t\right)$ and $\varphi_i\left(\mathbf{u}_1, \mathbf{u}_2^{opt}(t), \mathbf{x}, t, \mathbb{T}\right)$ for $i = 1,2,...,k$ are convex functions with respect to the control component $\mathbf{u}_1$. The theorem condition (8.29) is equivalent to the local maximum principle presented as the following inequality

$$H\left(\alpha_0, \psi(\mathbf{x},t), \mathbf{u}_1, \mathbf{u}_2^{opt}(t), \mathbf{x}, t\right) \leq H\left(\alpha_0, \psi(\mathbf{x},t), \mathbf{u}_1^{opt}, \mathbf{u}_2^{opt}(t), \mathbf{x}, t\right)$$

for $\mathbf{u}_1 \in \mathbb{R}^1$. The function $H(\alpha_0, \psi(\mathbf{x},t), \mathbf{u}_1, \mathbf{u}_2(t), \mathbf{x}, t)$ is convex with respect to $\mathbf{u}_1$ for any arbitrary values of $\alpha_0 \geq 0$, $\psi \in \mathbb{R}^1$, $\mathbf{u}_2 \in \mathbb{R}^1$, $\mathbf{x} \in \mathbb{R}^n$, $t \in \mathbb{R}^1$, if for example $\Phi_0(\mathbf{u},\mathbf{x},t)$ is a convex function and and $\phi(\mathbf{u},\mathbf{x},t)$ is a linear function with respect to $\mathbf{u}_1$.

Next we separate the control constraints and denote them as

$$\varphi_i(\mathbf{u}_1(\mathbf{x},t), \mathbf{x}, t) \leq 0,$$

$$\varphi_j(\mathbf{u}_2(t), t) \leq 0,$$

where $i = 1,2,...,k_1$ and $j = k_1+1, k_1+2, ..., k$.

In this case, the complementarity slackness conditions take the form

$$m_i^{(\mathbf{u}_1)}(\mathbf{x},t)\varphi_i\left(\mathbf{u}_1^{opt}(\mathbf{x},t), \mathbf{x}, t\right) \leq 0$$

and

$$m_j^{(\mathbf{u}_2)}(t)\varphi_j\left(\mathbf{u}_2^{opt}(t), t\right) \leq 0,$$

for $i = 1,2,...,k_1$ and $j = k_1+1, k_1+2, ..., k$. The local maximum principle (8.29) and (8.30) is

$$\psi(\mathbf{x},t)\phi'_{\mathbf{u}_1}\left(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T}\right) - \alpha_0 \Phi'_{o\mathbf{u}_1}\left(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T}\right)$$
$$-\sum_{i=1}^{k_1} m_i^{(\mathbf{u}_1)}(\mathbf{x},t)\,\varphi'_{i\mathbf{u}_1}\left(\mathbf{u}^{opt}(x,t),\mathbf{x},t\right) = 0$$

and

$$\int_{\Omega}\left[\psi(\mathbf{x},t)\,\phi'_{\mathbf{u}_2}\left(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T}\right) -\right.$$
$$\left.-\alpha_0 \Phi'_{o\mathbf{u}_2}\left(\mathbf{u}^{opt}(x,t),\mathbf{x},t,\mathbb{T}\right)\right] d\mathbf{x}$$
$$-\int_{\Omega}\sum_{j=k_1+1}^{k} m_j^{(\mathbf{u}_2)}(t)\,\varphi'_{j\mathbf{u}_2}\left(\mathbf{u}^0(x,t),\mathbf{x},t\right) = 0.$$

For the following derivation of the bang-bang control, we introduce the notations

$$\begin{aligned}
\widetilde{\phi}(\mathbf{x},t) &= \phi'_{\mathbf{u}_1}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t\right),\\
\widetilde{\widetilde{\phi}}(\mathbf{x},t) &= \phi'_{\mathbf{u}_2}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t\right),\\
\widetilde{\Phi}_0(\mathbf{x},t) &= \Phi'_{0\mathbf{u}_1}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t\right),\\
\widetilde{\widetilde{\Phi}}_0(\mathbf{x},t) &= \Phi'_{0\mathbf{u}_2}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t\right),\\
\widetilde{q}(\mathbf{x},t) &= q'_{\mathbf{u}_1}\left(\mathbf{u}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}\right),\\
\widetilde{\widetilde{q}}(\mathbf{x},t) &= q'_{\mathbf{u}_2}\left(\mathbf{u}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}\right),
\end{aligned}$$

Furthermore it is supposed that the control depends only on time. Then the local maximum principle takes the form

$$\int_{\Omega}\left[\psi(\mathbf{x},t)\,\phi'_{\mathbf{u}_2}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}\right) -\right.$$
$$\left.-\alpha_0 \Phi'_{o\mathbf{u}_2}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}\right)\right] d\mathbf{x}$$
$$-\int_{\Omega}\sum_{j=k_1+1}^{k} m_j^{(\mathbf{u}_2)}(t)\,\varphi'_{j\mathbf{u}_2}\left(\mathbf{u}^{opt}(\mathbf{x},t),\mathbf{x},t\right) = 0.$$

Taking into account the assumption of the control linearity, the right-hand side of (8.10) is now written as

$$\phi\left(\mathbf{u}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}\right) = \mathbf{u}(\mathbf{x},t)\,\widetilde{\phi}(\mathbf{x},t). \tag{8.33}$$

The same is done for the right-hand side of (8.12)

$$q\left(\mathbf{u}(\mathbf{x},t),\mathbf{x},t,\mathbb{T}\right) = \mathbf{u}(\mathbf{x},t)\,\widetilde{q}(\mathbf{x},t,\mathbb{T}), \tag{8.34}$$

and finally

$$\Phi_0\left(\mathbf{u}\left(\mathbf{x},t\right),\mathbf{x},t,\mathbb{T}\right)=\mathbf{u}\left(\mathbf{x},t\right)\widetilde{\Phi}_0\left(\mathbf{x},t,\mathbb{T}\right). \tag{8.35}$$

Now we consider two cases in order to get a bang-bang control problem.

Case 1. *The control vector* $\mathbf{u}\left(\mathbf{x},t\right)$ *has only one component* $u_1\left(\mathbf{x},t\right)$ and $\underline{u}_1 \leq u_1\left(\mathbf{x},t\right) \leq \overline{u}_1$. With respect to the limitations on the inductor mentioned above the constraints (8.13) are

$$-\mathbf{u}\left(\mathbf{x},t\right)+\underline{u}_1 \leq 0,$$

$$\mathbf{u}\left(\mathbf{x},t\right)-\overline{u}_1 \leq 0, \tag{8.36}$$

where $\mathbf{u} \in \mathbb{U}^{ad}$. We rewrite again the complementarity slackness as

$$\begin{aligned} m_1^{(u_1)}\left(\mathbf{x},t\right)\left(-\mathbf{u}\left(\mathbf{x},t\right)+\underline{u}_1\right) &= 0,\, m_1^{(u_1)}\left(\mathbf{x},t\right) \geq 0, \\ m_2^{(u_1)}\left(\mathbf{x},t\right)\left(\mathbf{u}\left(\mathbf{x},t\right)-\overline{u}_1\right) &= 0,\, m_2^{(u_1)}\left(\mathbf{x},t\right) \geq 0, \end{aligned} \tag{8.37}$$

getting a new expression for the local maximum principle

$$\begin{aligned} &\psi\left(\mathbf{x},t\right)\left(\widetilde{\phi}\left(\mathbf{x},t\right)-\widetilde{q}(\mathbf{x},t,\mathbb{T})\right)-\alpha_0\widetilde{\Phi}_0\left(\mathbf{x},t,\mathbb{T}\right)= \\ &-m_1^{(u_1)}\left(\mathbf{x},t\right)+m_2^{(u_1)}\left(\mathbf{x},t\right) \end{aligned} \tag{8.38}$$

for any $\left(\mathbf{x},t\right) \in \mathbb{U}$, where $\mathbb{U}$ is the set of admissible control.

Now, we have to study the equation (8.38). Let there be a domain $\left(\mathbf{x},t\right) \in \mathbb{U}$, where two inequalities are fulfilled, namely

$$\psi\left(\mathbf{x},t\right)\left(\widetilde{\phi}\left(\mathbf{x},t\right)-\widetilde{q}(\mathbf{x},t,\mathbb{T})\right)-\alpha_0\widetilde{\Phi}_0\left(\mathbf{x},t,\mathbb{T}\right)>0 \tag{8.39}$$

and

$$\psi\left(\mathbf{x},t\right)\left(\widetilde{\phi}\left(\mathbf{x},t\right)-\widetilde{q}(\mathbf{x},t,\mathbb{T})\right)-\alpha_0\widetilde{\Phi}_0\left(\mathbf{x},t,\mathbb{T}\right)<0. \tag{8.40}$$

Clearly, taking into account (8.37) for (8.38) with respect to (8.39) we get

$$m_2^{(u_1)}\left(\mathbf{x},t\right)>0$$

and

$$m_1^{(u_1)}\left(\mathbf{x},t\right)=0.$$

Now, it is easy to show that $\left(\mathbf{u}\left(\mathbf{x},t\right)-\overline{\mathbf{u}}_1\right)=0$ and $\mathbf{u}\left(\mathbf{x},t\right)=\overline{u}_1$. The same can be done in the case of (8.40), leading to

$$m_1^{(u_1)}\left(\mathbf{x},t\right)>0,$$

and

$$m_2^{(u_1)}(\mathbf{x},t) = 0.$$

So, $(-\mathbf{u}(\mathbf{x},t) + \overline{u}_1) = 0$ gives $\mathbf{u}(\mathbf{x},t) = \overline{u}_1$. If

$$\psi(\mathbf{x},t)\left(\widetilde{\phi}(\mathbf{x},t) - \widetilde{q}(\mathbf{x},t,\mathbb{T})\right) - \alpha_0 \widetilde{\Phi}_0(\mathbf{x},t,\mathbb{T}) = 0$$

is satisfied we have the bang-bang control case with

$$\mathbf{u}(\mathbf{x},t) \in [\underline{u}_1, \overline{u}_1].$$

Case 2. *The control vector* $\mathbf{u}(\mathbf{x},t)$ *has only one component* $u_2(t)$ and $\underline{u}_2 \leq u_2(t) \leq \overline{u}_2$. The local maximum principle is

$$\begin{aligned} &\int\limits_{\Omega}\left[\psi(\mathbf{x},t)\left(\widetilde{\widetilde{\phi}}(\mathbf{x},t) - \widetilde{\widetilde{q}}(\mathbf{x},t,\mathbb{T})\right) - \right. \\ &\left. -\alpha_0 \widetilde{\widetilde{\Phi}}_0(\mathbf{x},t)\right] d\mathbf{x} = -m_1^{(u_2)}(t) + m_2^{(u_2)}(t) \end{aligned} \tag{8.41}$$

for any $t \in I$.

Now let

$$\int\limits_{\Omega}\left[\psi(\mathbf{x},t)\left(\widetilde{\widetilde{\phi}}(\mathbf{x},t) - \widetilde{\widetilde{q}}(\mathbf{x},t,\mathbb{T})\right) - \alpha_0 \widetilde{\widetilde{\Phi}}_0(\mathbf{x},t)\right] d\mathbf{x} > \mathbf{0}$$

or

$$\int\limits_{\Omega}\left[\psi(\mathbf{x},t)\left(\widetilde{\widetilde{\phi}}(\mathbf{x},t) - \widetilde{\widetilde{q}}(\mathbf{x},t,\mathbb{T})\right) - \alpha_0 \widetilde{\widetilde{\Phi}}_0(\mathbf{x},t)\right] d\mathbf{x} > \mathbf{0}.$$

Using the reasoning as in the previous case, it is not difficult to show that $\mathbf{u}(\mathbf{x},t) \in [\underline{u}_2, \overline{u}_2]$.

8.5 An Illustrative Example

To illustrate the bang-bang optimal control presented in the previous section, we use an example based on the description [3] choosing a duralumin (AA2024) slab with a rectangle cross section $x_1^{\max} \times x_2^{\max} = 1 \times 2\text{m}^2$, which is placed into an induction coil. The power supply generates an alternating current at 50 Hz. Here, we consider the case where the whole capacity of the induction coil is controlled.

According to [18], mathematical equations describing the electromagnetic part are based in the generalized case on Helmholtz's equation

$$\bigtriangledown \times \left(\rho \bigtriangledown \times \mathbb{H}\right) = -2\pi f j \mu \mathbb{H},\ \mathbb{H}(\mathbf{x}) = \mathbb{H}_{\Gamma}. \tag{8.42}$$

The right-hand side of (8.10) takes a form

$$\phi = \rho \left(\nabla \times \mathbb{H}\right) \left(\nabla \times \mathbb{H}\right)^{\overline{*}}. \tag{8.43}$$

The mathematical model in a rectangle cross-section slab case takes the form

$$\mathbb{H} = \mathbb{H}_{\Gamma} \left[1 + \left(\frac{\cosh kx_2}{\cosh kx_2^{\max}} - 1 \right) \left(1 - \frac{\cosh A_2 kx_1}{\cosh A_2 kx_1^{\max}} \right) \right]$$

or

$$\mathbb{H} = \mathbb{H}_{\Gamma} \left[1 + \left(\frac{\cosh kx_1}{\cosh kx_1^{\max}} - 1 \right) \left(1 - \frac{\cosh A_1 kx_2}{\cosh A_1 kx_2^{\max}} \right) \right],$$

where $\mathbb{H}_{\Gamma}$ is the boundary function, which allows to control the induction heating process, $k = \frac{1+j}{\Delta}$ $(\Delta = \sqrt{\frac{\rho}{\pi f \mu}})$

$$A_i = \sqrt{\frac{2}{3 - \frac{kx_i^{\max} \tanh^2 kx_i^{\max}}{kx_i^{\max} - \tanh kx_i^{\max}}}}, \; i = 1,2.$$

This gives a possibility to rewrite (8.43) as

$$\phi\left(x_1, x_2\right) = \rho \left(\frac{\partial \mathbb{H}}{\partial x_1} \frac{\partial \mathbb{H}^{\overline{*}}}{\partial x_1} + \frac{\partial \mathbb{H}}{\partial x_2} \frac{\partial \mathbb{H}^{\overline{*}}}{\partial x_2} \right).$$

It is clear that the electromagnetic equation (8.42) and the heat transfer equation (8.10) are interconnected by means of $\rho = \rho\left(\mathbb{T}\right)$.

The parameters of the object equation (8.10) and the required constraints are chosen on the basis of the technological process for the external environment temperature $\mathbb{T}_E = 293K$ (see Fig.8.1 and Fig.8.2). The numerical results for the optimal control of the whole capacity of the induction coil, the maximum error for the temperature profile and the final temperature profile are presented in Fig. 8.3 - Fig. 8.5.

Next, simulations were made once only for the lower bound and once only for the upper bound of the parameters $\rho(\mathbb{T}) \in [0.95\rho_E(\mathbb{T}), 1.05\rho_E(\mathbb{T})]$, $c(\mathbb{T}) \in [0.95c_E(\mathbb{T}), 1.05c_E(\mathbb{T})]$, and $\lambda(\mathbb{T}) \in [0.95\lambda_E(\mathbb{T}), 1.05\lambda_E(\mathbb{T})]$ (where the index E means the parameter values were received for the ambient temperature $\mathbb{T}_E = 293K$). The optimal control strategies for both cases are presented in Fig. 8.6. Comparing control actions one can see the difference caused by the parameter variation. That means that during the production process the technologist has to know how the uncertainties of the external environment can affect the induction heating to avoid undesired outcomes.

8.6 Conclusions

In this work, we studied the theoretical backgrounds of the optimal control for induction heating in the case that the control actions enter the system through bound-

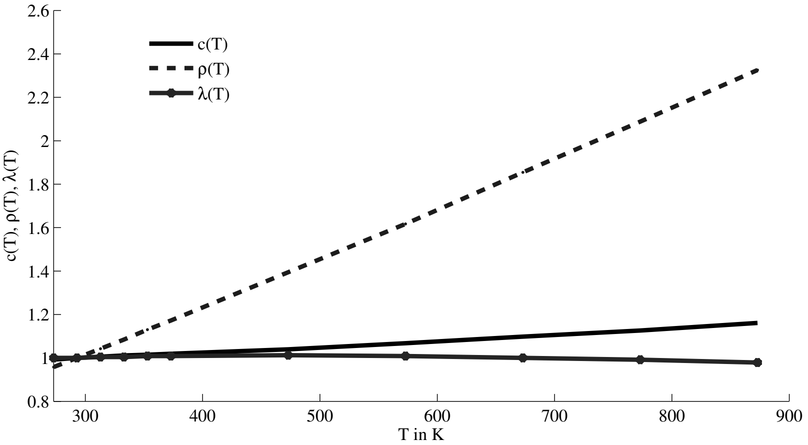


Fig. 8.1: The parameters of the technological process selected according to [12] and [17]

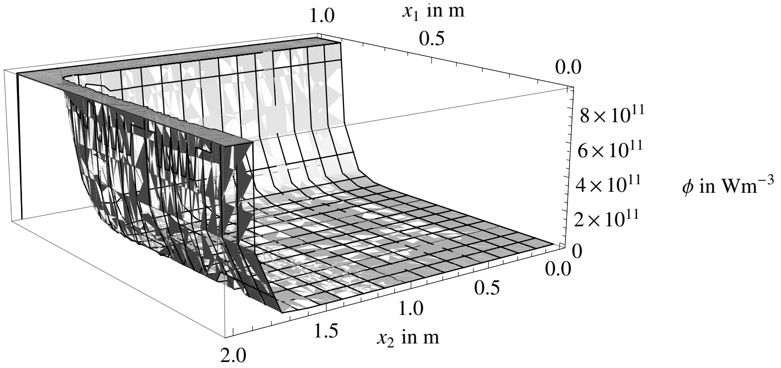


Fig. 8.2: The heat source protocol

ary conditions. The main result is presented as first-order necessary conditions. Numerical simulations showed that the theoretical results can be easily implemented in the case of the interval representation of the parameters. In further investigations, one can treat, for an example, ξ_1 and ξ_2 as some stochastic processes or use time-dependent distributions of parameters $\rho(\mathbb{T})$, $c(\mathbb{T})$, and $\lambda(\mathbb{T})$ in order to take the uncertainties of the technological processes in induction heating and to improve the quality of production.

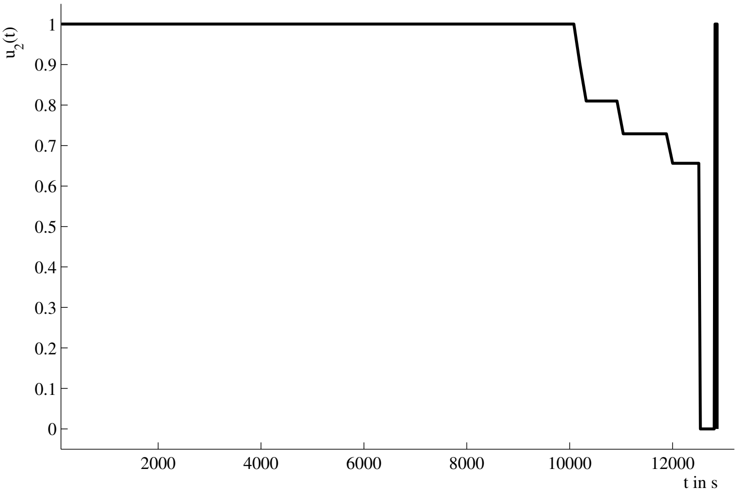


Fig. 8.3: The optimal control of the whole capacity of the induction coil (u_2 is presented in absolute units)

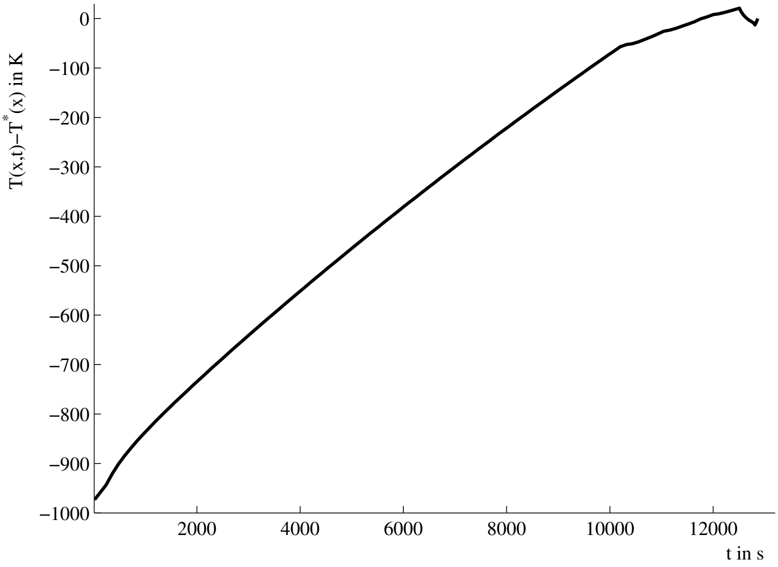


Fig. 8.4: The maximal error for temperature profile

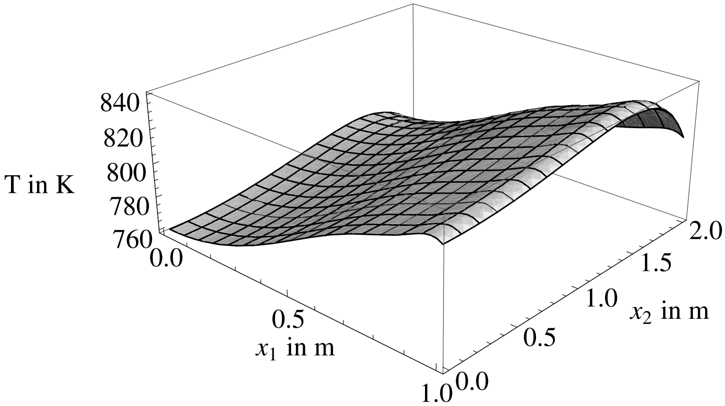


Fig. 8.5: Final temperature profile

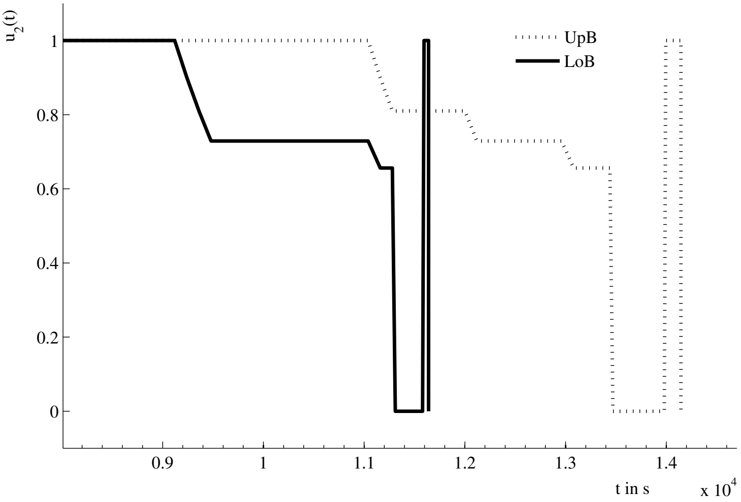


Fig. 8.6: The control protocol for low (LoB) and upper (UpB) bounds for the set of parameters ($u_2(t)$ is presented in absolute units)

Appendix

Nomenclature

$\mathbb{T}$	Temperature in K
c	(Heat capacity of workpiece)×(mass density) in $\mathrm{Jm^{-3}K^{-1}}$
λ	Thermal conductivity in $\mathrm{Wm^{-1}K^{-1}}$
q	Heat flux in $\mathrm{Wm^{-2}}$
ϕ	Heat source in $\mathrm{Wm^{-3}}$
ρ	Electrical resistivity in Ωm
μ	Magnetic permeability in $\mathrm{Wb^2N^{-1}m^{-2}}$
f	Frequency of electro-magnetic field in Hz
$\mathbb{H}$	Magnetic field intensity in $\mathrm{N\ Wb^{-1}}$
x_1, x_2	Space coordinates for workpiece in m
$x_1^{\max} \times x_2^{\max}$	Size of workpiece rectangular in cross-section in $\mathrm{m^2}$
∇	Nabla symbol
$\overline{*}$	Conjunction

References

1. Butkovskii, A.G., Malyi, S.A., Andreev, Ju.N.: Optimal control of heating metals. Metallurgia, Moscow (1972).
2. Butkovskii, A.G.: Structural theory of distributed systems. Nauka, Moscow (1997).
3. Dicoussar, V.V., Filatova, D.V., Grzywaczewski, M., Wójtowicz, M.: Optimal Control Coupled Fields in the Process of Induction Heating (Control Applications of Optimization 2003). Elsevier, Amsterdam (2003)
4. Favennec, Y., Rouizi, Y., Petit, D.: On the use of reduced models obtained through identification for feedback optimal control problems in a heat convection-diffusion problem. Comput. Methods Appl. Mech. Engrg **199**, 1193 – 1201 (2010)
5. Huang, M.-Sh., Huang, Y.-L.: Effect of multi-layered induction coils on efficiency and uniformity of surface heating. International Jounal of Heat and Mass Transfer **53**, 2414 – 2423 (2010)
6. Jang, J.-Y., Chiu, Y.W.: Numerical and experimental thermal analysis for a metalic hollow cylinder subjected to step-wise electro-magnitic induction heating. Applied Thermal Engineering **27**, 1883–1894 (2007)
7. Jiang, H., Nguyen, T.H., Prud'homme, M.: Optimal control of induction heating for semi-solid aluminum alloy forming. Journal of Materials Processing Technology **189**, 182 – 191 (2007)
8. Kantorovich, L.V., Akilov, G.P.: Functional analysis. Nauka, Moscow (1984)
9. Kranjc, M., Zupanic, A., Miklavic, D., Jarm, T.: Numerical analysis and thermographic investigation of induction heating. International Journal of Heat and Mass Transfer **53**, 3585 - 3591 (2010)
10. Milyutin, A.A., Osmolovskii, N.P.: Calculus of Variations and Optimal Control, *Translations of Mathematical Monographs*, volume 180, American Mathematical Society, Providence (1998)
11. Milyutin, A.A., Dmitruk, A.V., Osmolovskii, N.P.: Maximum Principle in Optimal Control. Moscow State University, Moscow (2004)

12. Okman, O., Dursunkaya, Z., Tekkaya, A.E.: Generalized transient temperature behavior in induction heated workpieces. Journal of Materials Processing Technology **209**, 5932 - 5939 (2009)
13. Padhi, R., Ali, S.F.: An account of chronological developments in control of distributed parameter systems. Annual Reviews in Control **33**, 59–68 (2009).
14. Rapoport, E.: Alternance method in applied tasks of optimization. Nauka, Moscow (2000)
15. Rapoport, E., Pleshivtseva, Yu.: Optimal Control of Induction Heating Processes, CRC Pr I Llc, Boston (2006)
16. Rapoport, E., Pleshivtseva, Yu.E.: Algorithmically precise method of parametric optimization in boundary-value optimal control problems for distributed-parameter systems. Optoelectronocs, Instruments and Data Processing **45 (5)**, 464 – 471 (2009)
17. Tslaf, A.: Combined properties of conductors and calculation of thermal processes in electrical and heat engineering. Elsevier, Boston (1981)
18. Zimin, L.S.: Heating peculiarities of rectangle bodies. Mashynostroyenie, Leningrad (1973)

Chapter 9
Coherent Upper and Lower Conditional Previsions Defined by Hausdorff Outer and Inner Measures

Serena Doria

Abstract A new model of coherent upper conditional previsions is proposed to represent uncertainty and to make previsions in complex systems. It is defined by the Choquet integral with respect to Hausdorff outer measure if the conditioning event has positive and finite Hausdorff outer measure in its Hausdorff dimension. Otherwise, when the conditioning event has Hausdorff outer measure equal to zero or infinity in its Hausdorff dimension, it is defined by a 0-1 valued finitely, but not countably, additive probability. If the conditioning event has positive and finite Hausdorff outer measure in its Hausdorff dimension, it is proven that a coherent upper conditional prevision is uniquely represented by the Choquet integral with respect to the upper conditional probability defined by Hausdorff outer measure if and only if it is monotone, comonotonically additive, submodular and continuous from below. Moreover sufficient conditions are given such that the upper conditional previsions satisfy the disintegration property and the conglomerability principle.

9.1 Introduction

Many complex systems are strongly dependent on the initial conditions, that is small differences on the initial conditions lead the system to entirely different states. These systems are called *chaotic* systems. Thus uncertainty in the initial conditions produces uncertainty in the final state of the system. Often the final state of the system is represented by a fractal set, i.e., a set with non-integer Hausdorff dimension. In this paper a new model of coherent upper and lower conditional previsions defined by Hausdorff outer and inner measures is proposed to represent uncertainty and to make previsions in complex systems. Coherent upper and lower conditional probabilities are obtained when only 0-1 valued random variables are considered. In [12]

Serena Doria
Department of Sciences, University G. D'Annunzio, Chieti-Pescara, Italy
e-mail: s.doria@dst.unich.it

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_9,

stochastic independence for fractal sets has been investigated with respect to coherent upper and lower conditional probabilities defined by Hausdorff outer and inner measures.

Coherent conditional previsions and probabilities are tools to model and quantify uncertainties; they have been investigated in de Finetti [15], [16], Dubins [13] Regazzini [25], [26] and Williams [35]. Upper and lower conditional previsions have been introduced in Walley [33], [34] and models of upper and lower conditional previsions have been analyzed in Vicig et al. [32] and Miranda and Zaffalon [22].

Coherent upper conditional previsions are functionals on a linear space of bounded random variables satisfying the axioms of coherence. They cannot always be defined as an extension of expectation of measurable random variables defined by the Radon-Nikodym derivative, according to the axiomatic definition. It occurs because one of the defining properties of the Radon-Nikodym derivative, that is to be measurable with respect to the σ-field of the conditioning events, contradicts a necessary condition for coherence. So the necessity to find a new mathematical tool in order to define coherent upper conditional previsions arises. In the subjective probabilistic approach coherent probability is defined on an arbitrary class of sets and any coherent probability can be extended to a larger domain. So in this framework no measurability condition is required for random variables. In the sequel, bounded random variables are bounded real-valued functions (these functions are called *gambles* in Walley [34] or *random quantities* in de Finetti [15]). When a measurability condition for a random variable is required, for example to define the Choquet integral, it is explicitly mentioned through the paper. In particular, various conditions of measurability proposed in the literature are considered:

i) *upper μ-measurability* of a random variable with respect to a monotone set function μ defined on a class S of sets, which requires that the decreasing distribution functions of a random variable with respect to the outer and inner measures generated by μ are equal [8].

ii) *upper S-measurability* of a random variable, which requires that the random variable is upper μ-measurable with respect to every monotone set function μ defined on S [8, p.49]; upper S-measurability implies that the Choquet integral with respect to μ depends only on the values that μ assumes on S.

iii) measurability of a random variable with respect to a partition **B** [34, p.291], which requires that the random variable is constant on the atoms of the partition.

iv) measurability of a random variable with respect to a σ-field , which requires that the preimage of every Borelian set of real numbers belongs to the σ-field.

If S is closed under intersection, upper S-measurability implies measurability of the random variable with respect to the partition of atoms of S [8, p.52]. If S is a σ-field and X and $-X$ are upper S-measurable then upper S-measurability of X is equivalent to the measurability of X with respect to the σ-field S.

Let (Ω,d) be a metric space and let **B** be a partition of Ω such that each $B \in \mathbf{B}$ is measurable with respect to the Hausdorff dimensional outer measure in its Hausdorff dimension. For every bounded random variable X defined on B a coherent upper conditional prevision $\overline{P}(X|B)$ is defined by the Choquet integral with respect to its associated Hausdorff outer measure if the conditioning event has positive and finite Hausdorff outer measure in its Hausdorff dimension. Otherwise if the conditioning event has Hausdorff outer measure in its Hausdorff dimension equal to zero or infinity it is defined by a 0-1 valued finitely, but not countably, additive probability. If the conditioning event B has positive and finite Hausdorff outer measure in its Hausdorff dimension then the given upper conditional prevision defined on the linear space of all bounded random variables on B is proven to be a functional, which is monotone, submodular, comonotonically additive and continuous from below. Moreover, given a class S of subsets of B, all these properties are proven to be a sufficient condition under which the upper conditional probability defined by Hausdorff outer measure is the unique monotone set function, which represent a coherent upper conditional prevision of upper S-measurable bounded random variables as Choquet integral. The paper is organized as follows. In section 2 the notion of coherent upper conditional prevision and its properties are recalled. In section 3 it is proven that conditional expectation, defined by the Radon-Nikodym derivative may be not coherent. A new model of upper conditional prevision defined with respect to Hausdorff outer measure is proposed in section 4. In section 5 a coherent upper conditional prevision is characterized as Choquet integral with respect to upper conditional probability defined by Hausdorff outer measure. In section 6 sufficient conditions are given such that the given upper conditional previsions satisfy the disintegration property and the conglomerability principle.

9.2 Separately Coherent Upper and Lower Conditional Previsions

Separately coherent upper conditional previsions $\overline{P}(\cdot|B)$ are functionals, defined on a linear space of bounded random variables, satisfying the axioms of separate coherence [34].

Definition 9.1. Let (Ω,d) be a metric space and let $\mathbf{B}$ be a partition of Ω. For every $B \in \mathbf{B}$ let $\mathbf{K}(B)$ be a linear space of bounded random variables defined on B. Let us denote by $X|B$ a random variable defined on B, or more generally the restriction to B of a random variable defined on Ω and by $\sup(X|B)$ the supremum value that X assumes on B. Then separately coherent upper conditional previsions are functionals $\overline{P}(\cdot|B)$ defined on $\mathbf{K}(B)$, such that the following conditions hold for every X and Y in $\mathbf{K}(B)$ and every strictly positive constant λ:

1) $\overline{P}\,(X|B) \leq \sup(X|B)$;
2) $\overline{P}(\lambda\ X|B) = \lambda\ \overline{P}(X|B)$ (positive homogeneity);
3) $\overline{P}(X+Y)|B) \leq \overline{P}(X|B) + \overline{P}(Y|B)$;
4) $\overline{P}(B|B) = 1$.

Separately coherent upper conditional previsions can always be extended to coherent upper previsions on the class $\mathbf{L}(B)$ of all bounded random variables defined on B.

Suppose that $\overline{P}(X|B)$ is a coherent upper conditional prevision on a linear space $\mathbf{K}(B)$ then its conjugate coherent lower conditional prevision is defined by $\underline{P}(X|B) = -\overline{P}(-X|B)$. If for every X belonging to $\mathbf{K}(B)$ we have $P(X|B) = \underline{P}(X|B) = \overline{P}(X|B)$ then $P(X|B)$ is called a coherent *linear* conditional prevision (de Finetti [15]) and it is a linear positive functional on $\mathbf{K}(B)$.

Definition 9.2. Let (Ω, d) be a metric space and let $\mathbf{B}$ be a partition of Ω. For every $B \in \mathbf{B}$ let $\mathbf{K}(B)$ be a linear space of bounded random variables defined on B. Then linear separately coherent conditional previsions are functionals $P(\cdot|B)$ defined on $\mathbf{K}(B)$, such that the following conditions hold for every X and Y in $\mathbf{K}(B)$ and every strictly positive constant λ:
1') if $X \geq 0$ then $P(X|B) \geq 0$ (positivity);
2') $P(\lambda X|B) = \lambda P(X|B)$ (positive homogeneity);
3') $P(X+Y)|B) = P(X|B) + P(Y|B)$ (linearity);
4') $P(B|B) = 1$.

A class of bounded random variables is called a *lattice* if it is closed under point-wise maximum $\vee$ and point-wise minimum $\wedge$.

Two random variables X and Y defined on B are *comonotonic* if, $(X(\omega_1) - X(\omega_2))(Y(\omega_1) - Y(\omega_2)) \geq 0 \; \forall \omega_1, \omega_2 \in B$.

Definition 9.3. Let (Ω, d) be a metric space and let $\mathbf{B}$ be a partition of Ω. For every $B \in \mathbf{B}$ let $\mathbf{K}(B)$ be a linear lattice of bounded random variables defined on B and let $\overline{P}(\cdot|B)$ be a coherent upper conditional prevision defined on $\mathbf{K}(B)$ then for every X, Y, X_n in $\mathbf{K}(B)$ $\overline{P}(\cdot|B)$ is

i) *monotone* iff $X \leq Y$ implies $\overline{P}(X|B) \leq \overline{P}(Y|B)$;
ii) *comonotonically additive* iff $\overline{P}(X+Y|B) = \overline{P}(X|B) + \overline{P}(Y|B)$ if X and Y are comonotonic;
iii)*submodular* iff $\overline{P}(X \vee Y|B) + \overline{P}(X \wedge Y|B) \leq \overline{P}(X|B) + \overline{P}(Y|B)$;
iv)*continuous from below* iff $lim_{n \to \infty} \overline{P}(X_n|B) = \overline{P}(X|B)$ if X_n is an increasing sequence of random variables converging to X.

A bounded random variable is called $\mathbf{B}$-measurable or measurable with respect to the partition $\mathbf{B}$ [34, p.291] if it is constant on the atoms B of the partition. Let $G(\mathbf{B})$ be the class of all $\mathbf{B}$-measurable random variables.

Denote by $\overline{P}(X|\mathbf{B})$ the random variable equal to $\overline{P}(X|B)$ if $\omega \in B$.

Separately coherent upper conditional previsions $\overline{P}(X|B)$ can be extended to a common domain $\mathbf{H}$ so that the function $\overline{P}(\cdot|\mathbf{B})$ can be defined from $\mathbf{H}$ to $G(\mathbf{B})$ to summarize the collection of $\overline{P}(X|B)$ with $B \in \mathbf{B}$.

$\overline{P}(\cdot|\mathbf{B})$ is assumed to be separately coherent if all the $\overline{P}(\cdot|B)$ are separately coherent. In the next section the function $\overline{P}(X|\mathbf{B})$ is compared with the Radon-Nikodym derivative.

9.3 The Radon-Nikodym Derivative May Fail to be Coherent

In [34, 6.5.8] a comparison between Kolmogorov theory and coherent linear previsions has been performed. The author points out that when the conditioning event has probability zero Kolmogorov allows conditional previsions to be completely arbitrary. In this section the role of the Radon-Nikodym derivative in the assessment of coherent linear prevision is analysed. Let Ω be a non-empty set; a class $\mathbf{F}$ of subsets of Ω is a *σ-field* if it contains Ω, it is closed under the formation of complements and countable unions. Let P be a probability measure on $\mathbf{F}$, that is P is a non-negative real function on $\mathbf{F}$, which is countably additive and such that $P(\Omega) = 1$. The triple $(\Omega, \mathbf{F}, P)$ is called a *probability space*.

We prove that when Ω is equal to [0,1] and the σ-field of the conditioning events is properly contained in the σ-field of the given probability space and contains all singletons of Ω, then conditional expectation may fail to be coherent.

In the axiomatic approach [3] conditional expectation is defined with respect to a σ-field $\mathbf{G}$ of conditioning events by the Radon-Nikodym derivative;

Definition 9.4. Let $\mathbf{F}$ and $\mathbf{G}$ be two σ-fields of subsets of Ω with $\mathbf{G}$ contained in $\mathbf{F}$ and let X be an integrable random variable. Let P be a probability measure on $\mathbf{F}$; define a measure ν on $\mathbf{G}$ by $\nu(G) = \int_G X dP$. This measure is finite and absolutely continuous with respect to P. Thus there exists a function, the *Radon-Nikodym derivative* denoted by $E[X|\mathbf{G}]$, defined on Ω, $\mathbf{G}$-measurable, integrable and satisfying the functional equation

$$\int_G E[X|\mathbf{G}]dP = \int_G X dP$$

with G in $\mathbf{G}$.

This function is unique up to a set of P-measure zero and it is a version of the conditional expected value.

The definitions of conditional expectation and coherent linear conditional prevision can be compared when the σ-field $\mathbf{G}$ is generated by the partition $\mathbf{B}$. In particular, given a probability space $(\Omega, \mathbf{F}, P)$, let $\mathbf{G}$ be equal or contained in the σ-field generated by a countable class $\mathbf{C}$ of subsets of $\mathbf{F}$ and let $\mathbf{B}$ be the partition generated by the class $\mathbf{C}$. Denote Ω' = $\mathbf{B}$, $P(A|\mathbf{B})$ the random variable equal to $P(X|B)$ if $\omega \in B$ and φ_B the function from Ω to Ω' that associates to every $\omega \in \Omega$ the atom B of the partition $\mathbf{B}$ that contains ω. Then we have that $P(A|\mathbf{G}) = P(A|B) \circ \varphi_B$ for every $A \in \mathbf{F}$ [20, p.262], that is $P(A|\mathbf{G}) = P(A|\mathbf{B})$.

The next theorem shows that every time that the σ-field $\mathbf{G}$ of the conditioning events is properly contained in $\mathbf{F}$ and it contains all singletons of [0,1] then the conditional prevision, defined by the Radon-Nikodym derivative is not coherent. It occurs because one of the defining properties of conditional expectation that is to be measurable with respect to the σ-field of conditioning events contradicts the following necessary condition for coherence of a linear conditional prevision. If $P(X|\mathbf{B})$ is separately coherent and X is $\mathbf{B}$-measurable then $P(X|\mathbf{B}) = X$ [34, p.292].

This necessary condition for coherence is not always satisfied if $P(X|\mathbf{B})$ is defined by the Radon-Nikodym derivative.

Theorem 9.1. *Let Ω = [0,1], let $\boldsymbol{F}$ be the Borel σ-field of [0,1] and let P be the Lebesgue measure on $\boldsymbol{F}$. Let $\boldsymbol{G}$ be a sub σ-field properly contained in $\boldsymbol{F}$ and containing all singletons of [0,1]. Let $\boldsymbol{B}$ be the partition of all singletons of [0,1] and let X be the indicator function of an event A belonging to $\boldsymbol{F}$ - $\boldsymbol{G}$. If $P(X|\boldsymbol{B})$ is equal to the Radon-Nikodym derivative with probability 1, that is*

$$P(X|\boldsymbol{B}) = E[X|\boldsymbol{G}]$$

except on a subset N of [0,1] of P-measure zero, then the function $P(X|\boldsymbol{B})$ is not separately coherent.

Proof. If the equality $P(X|\mathbf{B}) = E[X|\mathbf{G}]$ holds with probability 1, then we have that, with probability 1, the function $P(X|\mathbf{B})$ is different from X, the indicator function of A; in fact having fixed A in $\mathbf{F} - \mathbf{G}$ the indicator function X is not $\mathbf{G}$-measurable, it does not verify a property of the Radon-Nikodym derivative and therefore it cannot be assumed as conditional expectation according to the axiomatic definition. Therefore $P(X|\mathbf{B})$ does not satisfy the necessary condition for being coherent, $P(X|\mathbf{B}) = X$. □

Example 9.1. [3, Example 33.11] Let Ω = [0,1], let $\mathbf{F}$ be the Borel σ-field of Ω, let P be the Lebesgue measure on $\mathbf{F}$ and let $\mathbf{G}$ be the sub σ-field of $\mathbf{F}$ of sets that are either countable or co-countable. Let $\mathbf{B}$ be the partition of all singletons of Ω; if the linear conditional prevision is defined equal, with probability 1, to conditional expectation defined by the Radon-Nikodym derivative, we have that

$$P(X|\mathbf{B}) = E[X|\mathbf{G}] = P(X).$$

So when X is the indicator function of an event $A = [a,b]$ with $0 < a < b < 1$ then $P(X|\mathbf{B}) = P(A)$ and it does not satisfy the necessary condition for coherence that is $P(X|\mathbf{B}) = X$.

Evident from Theorem 1 and Example 1 is the necessity to introduce a new tool to define coherent conditional previsions.

9.4 Coherent Upper Conditional Previsions Defined by Hausdorff Outer Measures

In this section a coherent upper conditional prevision $\overline{P}(\cdot|B)$ is defined by the Choquet integral with respect to its associated Hausdorff outer measure if the conditioning event B has positive and finite Hausdorff outer measure in its Hausdorff dimension. Otherwise if the conditioning event B has Hausdorff outer measure in its Hausdorff dimension equal to zero or infinity it is defined by a 0-1 valued finitely, but not countably, additive probability. Let (Ω, d) be a metric space and let $\mathbf{B}$ be a partition of Ω. In the sequel each set B of $\mathbf{B}$ is required to be measurable with respect to h^s, the Hausdorff outer measure associated with B, so that the indicator function of each set B is h^s-upper measurable and $P(B|B)$ can be represented as the Choquet integral with respect to h^s.

Definition 9.5. Let (Ω, d) be a metric space and let $\mathbf{B}$ be a partition of Ω. For each B in $\mathbf{B}$, denote by s the Hausdorff dimension of B then the Hausdorff s-dimensional outer measure is called the Hausdorff outer measure *associated* with the coherent upper prevision $\overline{P}(\cdot|B)$. Let $B \in \mathbf{B}$ be measurable with respect to the Hausdorff outer measure associated with $\overline{P}(\cdot|B)$.

9.4.1 Hausdorff Outer Measures

Given a non-empty set Ω, let $\wp(\Omega)$ be the class of all subsets of Ω. An *outer measure* is a function $\mu^* : \wp(\Omega) \to [0,+\infty]$ such that $\mu^*(\oslash) = 0$, $\mu^*(A) \leq \mu^*(A')$ if $A \subseteq A'$ and $\mu^*(\bigcup_{i=1}^{\infty} A_i) \leq \sum_{i=1}^{\infty} \mu^*(A_i)$.

Examples of outer set functions or outer measures are the Hausdorff outer measures [14], [27].

Let (Ω, d) be a metric space. A topology, called the *metric topology*, can be introduced into any metric space by defining the open sets of the space as the sets G with the property:

if x is a point of G, then for some $r > 0$ all points y with $d(x,y) < r$ also belong to G.

It is easy to verify that the open sets defined in this way satisfy the standard axioms of the system of open sets belonging to a topology [27, p.26].

The diameter of a non empty set U of Ω is defined as $|U| = \sup\{d(x,y) : x, y \in U\}$ and if a subset A of Ω is such that $A \subset \bigcup_i U_i$ and $0 < |U_i| < \delta$ for each i, the class $\{U_i\}$ is called a δ-cover of A.

Let s be a non-negative number. For δ >0 we define $h_{s,\delta}$ (A) = inf $\sum_{i=1}^{+\infty} |U_i|^s$, where the infimum is over all δ-covers $\{U_i\}$.

The *Hausdorff s-dimensional outer measure* of A, denoted by $h^s(A)$, is defined as

$$h^s(A) = \lim_{\delta \to 0} h_{s,\delta}(A).$$

This limit exists, but may be infinite, since $h_{s,\delta}(A)$ increases as δ decreases because less δ-covers are available. The *Hausdorff dimension* of a set A, $\dim_H(A)$, is defined as the unique value, such that

$$h^s(A) = +\infty \text{ if } 0 \leq s < \dim_H(A),$$

$$h^s(A) = 0 \text{ if } \dim_H(A) < s < +\infty.$$

We can observe that if $0 < h^s(A) < +\infty$ then $dim_H(A) = s$, but the converse is not true. Hausdorff outer measures are *metric* outer measures ($h^s(E \cup F) = h^s(E) + h^s(F)$ whenever $d(E,F) = \inf\{d(x,y) : x \in E, y \in F\} > 0$).

A subset A of Ω is called *measurable* with respect to the outer measure h^s if it decomposes every subset of Ω additively, that is if $h^s(E) = h^s(A \cap E) + h^s(E - A)$ for all sets $E \subseteq \Omega$.

All Borel subsets of Ω are measurable with respect to any metric outer measure [14, Theorem 1.5]. So every Borel subset of Ω is measurable with respect to every Hausdorff outer measure h^s since Hausdorff outer measures are metric.

The restriction of h^s to the σ-field of h^s-measurable sets, containing the σ-field of the Borel sets, is called Hausdorff s-dimensional measure. The Borel σ-field is the σ-field generated by all open sets. The Borel sets include the closed sets (as complement of the open sets), the F_σ-sets (countable unions of closed sets) and the G_σ-sets (countable intersections of open sets), etc.

In particular the Hausdorff 0-dimensional measure is the counting measure and the Hausdorff 1-dimensional measure is the Lebesgue measure.

The Hausdorff s-dimensional measures are *modular* on the Borel σ-field, that is $h^s(A \cup B) + h^s(A \cap B) = h^s(A) + h^s(B)$ for every pair of Borelian sets A and B; so that [8, Proposition 2.4] the Hausdorff outer measures are *submodular* ($h^s(A \cup B) + h^s(A \cap B) \leq h^s(A) + h^s(B)$).

In [27, p.50] and [14, Theorem 1.6 (a)] it has been proven that if A is any subset of Ω there is a G_σ-set G containing A with $h^s(A) = h^s(G)$. In particular h^s is an *outer regular* measure.

Moreover Hausdorff outer measures are *continuous from below* [14, Lemma 1.3], that is for any increasing sequence of sets $\{A_i\}$ we have $\lim_{i \to \infty} h^s(A_i) = h^s(\lim_{i \to \infty} A_i)$.

h^s-Measurable sets with finite Hausdorff s-dimensional outer measure can be approximated from below by closed subsets [27, p.50] [14, Theorem 1.6 (b)] or equally the restriction of every Hausdorff outer measure h^s to the class of all h^s-measurable sets with finite Hausdorff outer measure is *inner regular* on the class of all closed subsets of Ω. In particular any h^s-measurable set with finite Hausdorff s-dimensional outer measure contains an F_σ-set of equal measure, and so contains a closed set differing from it by arbitrary small measure.

Since every metric space is a Hausdorff space then every compact subset of Ω is closed; denote by $\mathbf{O}$ the class of all open sets of Ω and by $\mathbf{C}$ the class of all compact sets of Ω, the restriction of each Hausdorff s-dimensional outer measure to the class $\mathbf{H}$ of all h^s-measurable sets with finite Hausdorff outer measure is *strongly regular* [8, p.43] that is it is regular:

r1)$h^s(A) = \inf\{h^s(U) | A \subset U, U \in \mathbf{O}\}$ for all $A \in \mathbf{H}$ (outer regular);
r2)$h^s(A) = \sup\{h^s(C) | C \subset A, C \in \mathbf{C}\}$ for all $A \in \mathbf{H}$ (inner regular)

with the additional property:

r3)$\inf\{h^s(U - A) | A \subset U, U \in \mathbf{O}\} = 0$ for all $A \in \mathbf{H}$

Any Hausdorff s-dimensional outer measure is translation invariant, that is, $h^s(x + E) = h^s(E)$, where $x + E = \{x + y : y \in E\}$ [14, p.18].

Hausdorff measure and dimension of Cantor sets were estimated in [5]. Hausdorff measure of the level sets of a Brownian motion was estimated in [24] and Hausdorff measure of arcs and Brownian motion was estimated in [7]. Hausdorff dimension and Hausdorff measure are considered in [4] to classify compact subsets of $\Re$ of Lebesgue measure zero.

9.4.2 The Choquet Integral

We recall the definition of the Choquet integral [8] with the aim of defining a coherent upper conditional prevision by Choquet integral with respect to Hausdorff outer measure and to prove its properties. The Choquet integral is an integral with respect to a monotone set function. Given a non-empty set Ω and denoting by S a set system, containing the empty set and properly contained in $\wp(\Omega)$, a monotone set function μ: $S \to \overline{\Re}_+ = \Re_+ \cup \{+\infty\}$ is such that $\mu(\oslash) = 0$ and if A, $B \in S$ with $A \subseteq B$ then $\mu(A) \leq \mu(B)$. Given a monotone set function μ on S, its *outer set function* is the set function μ^* defined on the whole power set $\wp(\Omega)$ by

$$\mu^*(A) = \inf\{\mu(B) : B \supset A; B \in S\}, A \in \wp(\Omega)$$

The inner set function of μ is the set function μ_* defined on the whole power set $\wp(\Omega)$ by

$$\mu_*(A) = \sup\{\mu(B)|B \subset A; B \in S\}, A \in \wp(\Omega)$$

Let μ be a monotone set function defined on S properly contained in $\wp(\Omega)$ and $X : \Omega \to \overline{\Re} = \Re \cup \{-\infty, +\infty\}$ an arbitrary function on Ω. Then the set function

$$G_{\mu,X}(x) = \mu\{\omega \in \Omega : X(\omega) > x\}$$

is decreasing and it is called *decreasing distribution function* of X with respect to μ. If μ is continuous from below then $G_{\mu,X}(x)$ is right continuous. In particular the decreasing distribution function of X with respect to the Hausdorff outer measures is right continuous since these outer measures are continuous from below. A function $X : \Omega \to \overline{\Re}$ is called *upper μ-measurable* if $G_{\mu^*,X}(x) = G_{\mu_*,X}(x)$. Given an upper μ-measurable function $X : \Omega \to \overline{R}$ with decreasing distribution function $G_{\mu,X}(x)$, if $\mu(\Omega) < +\infty$, the *asymmetric Choquet integral* of X with respect to μ is defined by

$$\int X d\mu = \int_{-\infty}^{0} (G_{\mu,X}(x) - \mu(\Omega))dx + \int_0^{\infty} G_{\mu,X}(x)dx$$

The integral is in $\Re$, can assume the values $-\infty$, $+\infty$ or is undefined when the right-hand side is $\infty - \infty$.

If $X \geq 0$ or $X \leq 0$ the integral always exists. In particular for $X \geq 0$ we obtain

$$\int X d\mu = \int_0^{+\infty} G_{\mu,X}(x)dx$$

If X is bounded and $\mu(\Omega)$ = 1 we have that

$$\begin{aligned}\int X d\mu &= \int_{\inf X}^{0} (G_{\mu,X}(x)-1)dx + \int_{0}^{\sup X} G_{\mu,X}(x)dx\\ &= \int_{\inf X}^{\sup X} G_{\mu,X}(x)dx + \inf X.\end{aligned}$$

9.4.3 A New Model of Coherent Upper Conditional Previsions

A new model of coherent upper conditional prevision is introduced and its properties are proven. Let (Ω, d) be a metric space and let $\mathbf{B}$ be a partition of Ω. For every $B \in \mathbf{B}$ denote by s the Hausdorff dimension of the conditioning event B, by h^s the Hausdorff s-dimensional outer measure and by h_s the Hausdorff s-dimensional inner measure.

Theorem 9.2. *Let B be an h^s-measurable set and let $\boldsymbol{K}(B)$ be a linear space of bounded random variables on B. Moreover, let m_B be a 0-1 valued finitely additive, but not countably additive, probability on $\wp(B)$ such that a different m_B is chosen for each B. Then for each $B \in \boldsymbol{B}$ the functional $\overline{P}(X|B)$ defined on $\boldsymbol{K}(B)$ by*

$$\overline{P}(X|B) = \tfrac{1}{h^s(B)} \int_B X dh^s \textit{ if } 0 < h^s(B) < +\infty$$

and by

$$\overline{P}(X|B) = m_B \textit{ if } h^s(B) = 0, +\infty$$

is a coherent upper conditional prevision.

Proof. Since $\mathbf{K}(B)$ is a linear space we have to prove that, for every $B \in \mathbf{B}$ $\overline{P}(X|B)$ satisfies conditions 1), 2), 3), 4) of Definition 1.

If B has finite and positive Hausdorff outer measure in its Hausdorff dimension s then $\overline{P}(X|B) = \frac{1}{h^s(B)} \int_B X dh^s$, so properties 1) and 2) are satisfied since they hold for the Choquet integral [8, Proposition 5.1]. Property 3) follows from the Subadditivity Theorem [8, Theorem 6.3] since Hausdorff outer measures are monotone, submodular and continuous from below. B is h^s - measurable then the indicator function of B is upper h^s - measurable and Property 4) holds since $\overline{P}(B|B) = \frac{1}{h^s(B)} \int_B dh^s = 1$. If B has Hausdorff outer measure in its Hausdorff dimension equal to zero or infinity, coherence requires that the restriction of a linear conditional prevision to events is a 0-1-valued finitely, but not countably, additive probability. Because linear previsions are uniquely determined by their restrictions to events thus the class of all coherent (upper) previsions on $\mathbf{L}(B)$ whose restrictions to events take only the values 0 and 1, can be identified with the class of 0-1-valued additive probabilities on $\wp(B)$. Then $\overline{P}(X|B) = m_B$ and properties 1), 2), 3) are satisfied since m_B is a 0-1 valued finitely, but not countably, additive probability on $\wp(B)$. Moreover since a different m_B is chosen for each B we have that $\overline{P}(B|B) = m_B(B) = 1$. □

The unconditional coherent upper prevision is obtained as a particular case when the conditioning event is Ω. Coherent upper conditional probabilities are obtained when only 0-1 valued random variables are considered; they have been defined in [9]:

Theorem 9.3. *Let m be a 0-1 valued finitely additive, but not countably additive, probability on $\wp(B)$ such that a different m is chosen for each B. Thus, for each $B \in \mathbf{B}$, the function defined on $\wp(B)$ by*

$$\overline{P}(A|B) = \tfrac{h^s(AB)}{h^s(B)} \text{ if } 0 < h^s(B) < +\infty$$

and by

$$\overline{P}(A|B) = m(AB) \text{ if } h^s(B) = 0, +\infty$$

is a coherent upper conditional probability.

Given an upper conditional prevision $\overline{P}(X|B)$ defined on a linear space the lower conditional prevision $\underline{P}(X|B)$ is obtained as its conjugate, that is $\underline{P}(X|B) = -\overline{P}(-X|B)$. If B has positive and finite Hausdorff outer measure in its Hausdorff dimension s and we denote by h_s the Hausdorff s-dimensional inner measure we have

$$\underline{P}(X|B) = -\overline{P}(-X|B) = -\frac{1}{h^s(B)}\int_B (-X)dh^s =$$
$$\frac{1}{h^s(B)}\int_B X dh_s = \frac{1}{h_s(B)}\int_B X dh_s.$$

The last equality holds since each B is h^s-measurable, that is $h^s(B) = h_s(B)$.

Let B be a set with positive and finite Hausdorff outer measure in its Hausdorff dimension s. Denote by h^s the s-dimensional Hausdorff outer measure and for every $A \in \wp(B)$ by $\mu_B^*(A) = \overline{P}(A|B) = \frac{h^s(AB)}{h^s(B)}$ the upper conditional probability defined on $\wp(B)$. From Theorem 2 we have that the upper conditional prevision $\overline{P}(\cdot|B)$ is a functional defined on $\mathbf{L}(B)$ with values in $\Re$ and the upper conditional probability μ_B^* integral represents $\overline{P}(X|B)$ since $\overline{P}(X|B) = \int X d\mu_B^* = \frac{1}{h^s(B)}\int X dh^s$. The number $\frac{1}{h^s(B)}$ is a normalizing constant.

In the following theorem it is proven that, if the conditioning event has positive and finite Hausdorff outer measure in its dimension s and $\mathbf{K}(B)$ is a linear lattice of bounded random variables defined on B, necessary conditions for the functional $\overline{P}(X|B)$ to be represented as Choquet integral with respect to the upper conditional probability μ_B^*, i.e. $\overline{P}(X|B) = \frac{1}{h^s(B)}\int X dh^s$, are that $\overline{P}(X|B)$ is monotone, comonotonically additive, submodular and continuous from below.

Theorem 9.4. *Let B be an h^s-measurable set and let $\boldsymbol{K}(B)$ be a linear lattice of bounded random variables defined on B. If the conditioning event B has positive and finite Hausdorff s-dimensional outer measure then the coherent upper conditional prevision $\overline{P}(\cdot|B)$ defined on $\boldsymbol{K}(B)$ as in Theorem 2 is:*

i) monotone;
ii) comonotonically additive;
iii)submodular;
iv) continuous from below.

Proof. Since the conditioning event B has positive and finite Hausdorff outer measure in its Hausdorff dimension s then the functional $\overline{P}(\cdot|B)$ is defined on $\mathbf{K}(B)$ by the Choquet integral with respect to the coherent upper conditional probability $\mu_B^*(A) = \frac{h^s(AB)}{h^s(B)}$; so conditions $i)$ and $ii)$ are satisfied because they are properties of the Choquet integral [8, Proposition 5.2].

Condition $iii)$ is equivalent to require that the monotone set function that represents the functional $\overline{P}(\cdot|B)$ is submodular and it is satisfied since Hausdorff outer measures are submodular. Moreover every s-dimensional Hausdorff measure is continuous from below then from the Monotone Convergence Theorem [8, Theorem 8.1] we have that the functional $\overline{P}(\cdot|B)$ is continuous from below, that is condition iv). □

9.4.4 Exactness and n-Monotonicity

In Maaß [21] exact functionals are introduced as a mathematical tool to unify the concepts of coherent lower previsions, exact cooperative games [29] and coherent risk measures [1].

Exact functionals are real-valued functionals that are monotone, superadditive, positively homogeneous and translation invariant (or constant additive) [21, Definition 1]. A dual theory can be derived for monotone, subadditive, positively homogeneous and constant additive functionals.

In Proposition 7 of [21] it has been proven that an exact functional is representable as Choquet integral if and only if it is comonotonically additive, moreover the Choquet integral with respect to a monotone set function μ is exact if and only if μ is supermodular. In the next theorem we prove that the lower conditional previsions obtained as the conjugate of the upper conditional previsions defined as in Theorem 2 are exact functionals.

Theorem 9.5. *Let $\boldsymbol{K}(B)$ be a linear space of bounded random variables on B. The lower conditional prevision $\underline{P}(\cdot|B)$ on $\boldsymbol{K}(B)$ obtained as the conjugate of the upper conditional prevision defined as in Theorem 2 is an exact functional on $\boldsymbol{K}(B)$.*

Proof. Let $\underline{P}(\cdot|B)$ be the lower conditional prevision obtained as the conjugate of the upper conditional prevision defined as in Theorem 2. From Proposition 7 of [21] we obtain that if B has positive and finite Hausdorff inner measure in its dimension then the functional $\underline{P}(X|B)$ is exact since Hausdorff inner measures are supermodular. If B has Hausdorff inner measure equal to zero or infinity in its dimension then the lower conditional prevision $\underline{P}(X|B)$ is a 0-1-valued finitely, but not countably, additive probability and so it is exact since it is a monotone linear functional on $\mathbf{K}(B)$. □

In de Cooman et al. [6] a special subclass of exact functionals, namely *n-monotone* are studied.

Definition 9.6. Let Γ be a functional whose domain is a lattice of bounded random variables on Ω. Γ is called n-monotone if for all $p \in N$, $p \leq n$, and all $f, f_1, ..., f_p$ belonging to the domain of Γ:

$$\sum_{I \subseteq \{1,...,p\}} (-1)^{|I|} \Gamma(f \wedge \bigwedge_{i \in I} f_i) \geq 0$$

The conjugate of an n-monotone functional is called *n-alternating*. An∞-monotone functional is also called *completely monotone* (i.e. it is a functional n-monotone for all $n \in N$) and its conjugate *completely alternating*.

An n-monotone functional on a lattice of events is called an *n-monotone* set function.

Theorem 9.6. *Let $\boldsymbol{K}(B)$ be a linear space of all bounded random variables on B. The lower conditional prevision $\underline{P}(\cdot|B)$ on $\boldsymbol{K}(B)$ obtained as the conjugate of the upper conditional prevision defined as in Theorem 2 is a completely monotone functional on $\boldsymbol{K}(B)$.*

Proof. If the conditioning event B has positive and finite Hausdorff s-dimensional inner measure in its dimension let $P(\cdot|B)$ be the restriction to the Borel σ-field of subsets of B, of the lower conditional probability $\underline{P}(\cdot|B)$, which is the conjugate of the upper conditional probability defined as in Theorem 3. Then $P(\cdot|B)$ is a linear exact set function on a lattice of events; thus [6, Theorem 5] it is always completely monotone and completely alternating. Moreover the lower conditional probability $\underline{P}(\cdot|B)$ is the natural extension to the class of all events of $P(\cdot|B)$ then [6, Theorem 7] $\underline{P}(\cdot|B)$ is also completely monotone. Since the Choquet functional with respect to an exact set function μ on $\wp(B)$ is n-monotone if and only if μ is [6, Theorem 9] then we obtain that the lower conditional prevision $\underline{P}(\cdot|B)$ is completely monotone because it is defined by the Choquet integral with respect to the completely monotone set function defined by the Hausdorff inner measure.

If the conditioning event B has Hausdorff s-dimensional inner measure equal to zero or infinity in its dimension then the lower conditional prevision $\underline{P}(\cdot|B)$ is completely monotone and completely alternating because it is defined by a 0-1 valued finitely, but not countably, additive probability. □

9.5 Uniqueness of the Representing Set Function for a Coherent Upper Conditional Prevision

If the conditioning event B has positive and finite Hausdorff measure in its Hausdorff dimension there is the problem of determining conditions, which assure that a coherent upper conditional prevision $\overline{P}(\cdot|B)$ can be represented by the Choquet integral with respect to a monotone set function and to determine the interval of monotone set functions which represent $\overline{P}(\cdot|B)$.

The representation of coherent lower previsions as Choquet integrals with respect to supermodular lower probabilities has been studied in [6]. In the quoted paper a representation result for exact n-monotone functionals in terms of Choquet integrals has been proven. The result does not address uniqueness of the representing function. The existence and uniqueness of a representation of coherent upper conditional previsions as Choquet integral with respect to Hausdorff outer measures has been studied in [11] [10]. Given a conditioning event with positive and finite Hausdorff outer measure in its dimension, it is proven that a coherent upper conditional prevision is monotone, submodular, comonotonically additive and continuous from below if and only if it is uniquely representable as the Choquet integral with respect to the coherent upper conditional probability defined by its associated Hausdorff outer measure.

Given a topological space (Ω, τ) conditions under which a functional can be represented as integral with respect to some monotone set functions are investigated in Greco [18], Bassanezi, Greco [2] Schmeidler [30], Denneberg [8] and Narukawa et al. [23].

Greco's Representation Theorem (Greco [18], Denneberg [8, Theorem 13.2]) assures that if Γ is a monotone and comonotonically additive functional defined on a linear lattice of bounded and unbounded random variables and such that it satisfies the upper and lower marginal continuity properties [8, (iv), (v) p. 156] then there exists a monotone set function defined on $\wp(\Omega)$ which represents Γ.

In Schmeidler [30] it has been proven that if a functional Γ, defined on a class of measurable bounded random variables, is monotone and comonotonically additive then it can be represented by the Choquet integral with respect to a monotone set function ν defined by $\nu(A) = \Gamma(A)$ on a set S.

In Narukawa et al. [23] it has been proven that a functional defined on the class of continuous functions with compact support, which is monotone and comonotonically additive is represented by the Choquet integral with respect to a unique regular fuzzy measure.

Given a family $\mathbf{L}$ of functions $X : \Omega \to \overline{\Re}$ and a functional $\Gamma : \mathbf{L} \to \overline{\Re}$ we say that Γ can be represented as Choquet integral with respect to a monotone set function μ on $\wp(\Omega)$ if $\Gamma(X) = \int X d\mu$. In Denneberg [8, Chapter 13], representation theorems for functionals with minimal requirements on the domain are examined. Let $\mathbf{L}$ be a class of random variables such that

a) $X \geq 0$ for all $X \in \mathbf{L}$ (non negativity);
b) $aX, X \wedge a, X - X \wedge a \in \mathbf{L}$ if $X \in \mathbf{L}$, $a \in \Re^+$;
c) $X \wedge Y, X \vee Y$ if $X, Y \in \mathbf{L}$ (lattice property).

In [8, Proposition 13.5] it is proven that if a functional Γ, defined on the domain $\mathbf{L}$, is monotone, comonotonically additive, submodular and continuous from below then Γ is representable as Choquet integral with respect to a monotone, submodular set function which is continuous from below. Furthermore all set functions on $\wp(\Omega)$ with these properties agree on the set system of weak upper level sets

$M = \{\{X \geq x\} \,|X \in \mathbf{L}, x \in \Re_{+}\}$. The uniqueness of the representing set function [8, Lemma 13.1] is due to the fact that the function $\Gamma(X \wedge x)$ determines the distribution function $G_{\mu,X}$ of an upper μ-measurable and positive random variable X with respect to any set function μ representing Γ; it occurs since $G_{\mu,X} = \frac{d}{dx}\Gamma(X \wedge x)$ for $X \in \mathbf{L}$ and for all $x \in \Re^{+}$ of continuity for $G_{\mu,X}$. If μ is continuous from below then $G_{\mu,X}$ is right continuous and it is the derivative from the right of $\Gamma(X \wedge x)$ for every $x \in \Re^{+}$. If the domain $\mathbf{L}$ is a linear lattice containing all constants this result can be extended to every bounded random variable. In fact since X is bounded, there exists a constant k such that $Y = X - k \in \mathbf{L}$ and $Y = X - k \geq 0$ so that $G_{\mu,Y} = \frac{d}{dx}\Gamma(Y \wedge x)$.

In the next theorem a sufficient condition is given such that a coherent upper conditional prevision is uniquely represented as Choquet integral with respect to the upper conditional probability μ_B^* defined by its associated Hausdorff outer measure. It is proven that if the conditioning event B has positive and finite Hausdorff outer measure in its Hausdorff dimension s and the coherent upper conditional prevision $\overline{P}(\cdot|B)$ is monotone, comonotonically additive, submodular and continuous from below then the upper conditional probability μ_B^* defined by the s-dimensional Hausdorff outer measure h^s is the unique monotone set function on the set system of weak upper level sets $M = \{\{X \geq x\} \,|X \in \mathbf{L}(B), x \in \Re\}$, which is submodular, continuous from below and representing $\overline{P}(\cdot|B)$ as Choquet integral. That is for every monotone set function β on $\wp(B)$, which is submodular, continuous from below and represents $\overline{P}(\cdot|B)$ we have that

$$\overline{P}(X|B) = \int_B X d\beta = \int_B X d\mu_B^* = \frac{1}{h^s(B)} \int_B X dh^s$$

for every bounded random variable X.

Theorem 9.7. *Let B be an h^s-measurable set and let $\boldsymbol{K}(B)$ be a linear lattice of bounded random variables on B containing all constants. If B has positive and finite Hausdorff outer measure in its dimension and the coherent upper conditional prevision $\overline{P}(\cdot|B)$ on $\boldsymbol{K}(B)$ is monotone, comonotonically additive, submodular and continuous from below then $\overline{P}(\cdot|B)$ is representable as Choquet integral with respect to a monotone, submodular set function which is continuous from below. Furthermore all monotone set functions on $\wp(B)$ with these properties agree on the set system of weak upper level sets $M = \{\{X \geq x\} \,|X \in \boldsymbol{K}(B), x \in \Re\}$ with the upper conditional probability $\mu_B^*(A) = \frac{h^s(AB)}{h^s(B)}$ for $A \in \wp(B)$. Let β be a monotone set function on $\wp(B)$, which is submodular, continuous from below and such that represents $\overline{P}(\cdot|B)$ as Choquet integral. Then the following equalities hold*

$$\overline{P}(X|B) = \int_B X d\beta = \int_B X d\mu_B^* = \frac{1}{h^s(B)} \int_B X dh^s.$$

Proof. $\mathbf{K}(B)$ is a linear lattice containing all constants so we can assume that property a) is true because otherwise since X is bounded there exists a constant k such that $X - k \in \mathbf{K}$(B) and $X - k \geq 0$. Moreover conditions b) and c) are satisfied. So from Proposition 13.5 of [8] we obtain that the functional $\overline{P}(\cdot|B)$ is representable by a monotone, submodular, continuous from below set function and all set functions with these properties agree on the set system of weak upper level sets $M = \{\{X \geq x\} | X \in \mathbf{K}(B), x \in \Re\}$. Every s-dimensional Hausdorff outer measure is monotone, submodular and continuous from below so, if B has positive and finite Hausdorff outer measure in its dimension then the monotone set function $\mu_B^*(A) = \frac{h^s(AB)}{h^s(B)}$ defined on $\wp(B)$ by the s-dimensional Hausdorff measure represents the functional $\overline{P}(\cdot|B)$. Moreover all monotone set functions on $\wp(B)$ which are submodular, continuous from below and represent the functional $\overline{P}(\cdot|B)$ agree on the set system of weak upper level sets with the upper conditional probability $\mu_B^*(A) = \frac{h^s(AB)}{h^s(B)}$. Denote by β any monotone set function on $\wp(B)$, which is submodular, continuous from below and such that it represents $\overline{P}(\cdot|B)$ as Choquet integral. Then μ_B^* and β agree on the set system of weak upper level sets M and $G_{\mu_B^*,X}(x) = G_{\beta,X}(x)$. μ_B^* and β represent the coherent upper conditional prevision $\overline{P}(\cdot|B)$ so we have that $\mu_B^*(B) = \beta(B) = 1$. Moreover since every X belonging to $\mathbf{K}(B)$ is bounded the following equalities hold:

$$\overline{P}(X|B) = \int_B X d\beta = \int_{\text{inf}X}^{\text{sup}X} G_{\beta,X}(x)dx + \text{inf}X =$$

$$\int_{\text{inf}X}^{\text{sup}X} G_{\mu_B^*,X}(x)dx + \text{inf}X = \int_B X d\mu_B^* = \frac{1}{h^s(B)} \int_B X dh^s$$

. □

The same result can be obtained if the coherent upper conditional probabilities μ_B^* and β are defined on a class S properly contained in $\wp(B)$ and $\mathbf{K}(B)$ is a linear lattice of bounded upper S-measurable random variables on B containing all constants.

Given a monotone set function β in Greco [17] a definition of measurability for positive functions with respect to a class S of subsets of Ω is given with the aim of determining the functions X such that the Choquet integral $\int X d\beta$ depends only on the values of β on S.

Definition 9.7. [17, p.165] A positive random variable X is *S-measurable* if and only if $\int X d\beta = \int X d\alpha$, where α, β are monotone set functions defined on $\wp(\Omega)$ such that $\alpha(A) = \beta(A)$ for every set A in S. A random variable X is S-measurable if X^+ and X^- are S-measurable where $X^+ = X \vee 0$ and $X^- = (-X) \vee 0$.

The previous definition is proven [17, Theorem 1] to be equivalent to the following condition 5):

$\forall a, b \in \Re, a < b$ there exists a set $H \in S$ so that $\{X > a\} \supset H \supset \{X > b\}$.

In Denneberg [8, p.49] a random variable X is defined to be *upper S-measurable* if it is upper μ-measurable $(G_{\mu^*,X}(x) = G_{\mu_*,X}(x))$ for any monotone set function μ on S. Condition 5) is a necessary and sufficient condition [8, Proposition 4.2] for upper S-measurability of a random variable X. In particular X is upper S-measurable if the upper set system $M_X = \{\{X \geq x\}, x \in \Re\}$, is contained in S. If S is a σ-field and M_X and M_{-X} are contained in S then we have the classical condition of measurability of functions.

Theorem 9.8. *For every $B \in \mathbf{B}$ let B be an h^s-measurable set and let $\mathbf{K}(B)$ be a linear lattice of bounded random variables on B containing all constants. Let $\overline{P}(\cdot|B)$ be a coherent upper conditional prevision which is monotone, submodular, comonotonically additive and continuous from below. Let S be a subclass properly contained in $\wp(B)$ such that it contains the set system of weak upper level sets $M = \{\{X \geq x\} | X \in \mathbf{K}(B); x \in \Re\}$. Denote by s the Hausdorff dimension of the conditioning event B and by h^s the Hausdorff s-dimensional outer measure. If $0 < h^s(B) < +\infty$ define $\mu_B^*(A) = \frac{h^s(AB)}{h^s(B)}$, for every $A \in S$ and let β be a coherent upper probability on S, which is submodular and continuous from below. Then the following equalities hold:*

$$\overline{P}(X|B) = \textstyle\int X d\beta = \int X d\mu_B^* = \frac{1}{h^s(B)} \int X dh^s.$$

Proof. $\mathbf{K}(B)$ satisfies the properties a), b) and c) since it is a linear lattice of bounded random variables on B containing all constants. Moreover $\overline{P}(\cdot|B)$ defined on $\mathbf{K}(B)$ is a coherent upper conditional prevision, which is monotone, submodular, comonotonically additive and continuous from below; S contains the set system of weak upper level sets $M = \{\{X \geq x\} X \in \mathbf{K}(B), x \in \Re\}$ then every bounded $X \in \mathbf{K}(B)$ is upper S-measurable, moreover since $\mathbf{K}(B)$ contains all constants then B belongs to S. μ_B^* and β are coherent upper probabilities on S so that $\mu_B^*(B) = \beta(B) = 1$; they are submodular, continuous from below and represent the functional $\overline{P}(\cdot|B)$ then [8, Proposition 13.5] they agree on the set system of weak upper level sets M and $G_{\mu_B^*,X}(x) = G_{\beta,X}(x)$. Moreover $\overline{P}(\cdot|B)$ is representable as Choquet integral with respect to μ_B^* and with respect to β. Then the following equalities hold:

$$\overline{P}(X|B) = \int_B X d\beta = \int_{\inf X}^{\sup X} G_{\beta,X}(x) dx + \inf X =$$

$$\int_{\inf X}^{\sup X} G_{\mu_B^*,X}(x) dx + \inf X = \int_B X d\mu_B^* = \frac{1}{h^s(B)} \int_B X dh^s.$$

□

The particular case where $\mathbf{K}(B)$ is the linear space of all bounded Borel-measurable random variables on B and S is the Borel σ-field of subsets of B is considered.

Example 9.2. Let $\mathbf{B}$ be a partition of Ω such that every $B \in \mathbf{B}$ is a Borelian set. For every $B \in \mathbf{B}$ let $\mathbf{K}(B)$ be the linear space of all bounded Borel-measurable random variables on B and let S be the Borel σ-field of subsets of B. Denote by s the Hausdorff dimension of the conditioning event B and by h^s the Hausdorff s-dimensional outer measure. If $0 < h^s(B) < +\infty$ define $\mu_B(A) = \frac{h^s(AB)}{h^s(B)}$, for every $A \in S$; $\mu_B(A)$ is modular and continuous from below on S since each Hausdorff s-dimensional (outer) measure is σ-additive on the Borel σ-field . Moreover let $P(\cdot|B)$ be a coherent linear conditional prevision, which is continuous from below. Then $P(\cdot|B)$ can be uniquely represented as the Choquet integral with respect to the coherent upper conditional probability μ_B, that is

$$P(X|B) = \textstyle\int X d\mu_B = \frac{1}{h^s(B)} \int X dh^s.$$

9.6 Coherence With Respect to the Unconditional Prevision

In this section the coherence with respect to the unconditional prevision of the upper conditional previsions, introduced in Theorem 2, is investigated. Walley [34, 6.3] discusses when an unconditional lower prevision $\underline{P}$ is coherent with the lower conditional prevision $\underline{P}(\cdot|\mathbf{B})$. In some special cases coherence of $\underline{P}$ and $\underline{P}(\cdot|\mathbf{B})$ can be characterized by simpler conditions. In particular in Walley [34, section 6.5.3 and section 6.5.7] it has been proven that if P and $P(\cdot|\mathbf{B})$ are respectively linear unconditional and conditional previsions on the same domain and $P(\cdot|\mathbf{B})$ are separately coherent, then P and $P(\cdot|\mathbf{B})$ are coherent if and only if the following *conglomerative property* is satisfied $P(X) = P(P(X|B))$. When the domains of P and $P(\cdot|\mathbf{B})$ are equal to the set of all bounded random variables on Ω then the conglomerative property is equivalent to the notion of *disintegrability* of a prevision $P(X)$ with respect to a partition of Ω, introduced by Dubins [13], which is equivalent to the *conglomerative principle* of de Finetti.

The relation between conglomerability and countable additivity has been investigated in [34, section 6.9] and [28]. In [28] it has been proven that when a probability P is defined at least on a σ-field and it assumes infinitely many different values then it is fully conglomerable if and only if it is countably additive on every partition of Ω. It means that we can find examples of merely additive probabilities defined on a field, that is not a σ-field, that assume only finitely many values and that are conglomerable with respect to a given partition [31, Example 5.5.]. But since every merely finitely additive probability defined on a field can be extended to a σ-field and to the power set, we have that every extension of this kind of probability to a σ-field is not fully conglomerable, since it fails conglomerability with respect to some countable partitions. We have that for non-countable partitions countable additivity of the unconditional prevision is not a sufficient condition to assure that it is coherent with the conditional previsions [19, Example 6.1].

In the next theorem sufficient conditions are given such that the upper conditional previsions defined as in Theorem 2 satisfy the disintegration property and the conglomerative principle. The result is based on the fact that if a random variable X is upper S-measurable then it is constant on the atoms of the partition of S [8, Example 4.4], i.e. X is $\mathbf{B}$-measurable [34] if $\mathbf{B}$ is the partition of atoms of S.

Theorem 9.9. *Let S be a subclass properly contained in $\wp(\Omega)$, such that it is closed under intersection and let $\boldsymbol{B}$ be the partition of atoms of S. For each $B \in \boldsymbol{B}$ denote by s the Hausdorff dimension of B and by h^s the Hausdorff s-dimensional outer measure. Let $\boldsymbol{K}(\Omega)$ be a linear space of bounded random variables on Ω, which are upper S-measurable. Then $\overline{P}(X|\boldsymbol{B})$ satisfies the disintegration property, i.e. $\overline{P}(X) = \overline{P}(\overline{P}(X|\boldsymbol{B}))$ and the conglomerability principle, i.e.* $\inf_{B\in\boldsymbol{B}}\overline{P}(X|B) \leq \overline{P}(X) \leq \sup_{B\in\boldsymbol{B}}\overline{P}(X|B)$.

Proof. Since X is upper S-measurable, then it is constant on the atoms of S, which are the sets of the partition $\mathbf{B}$ [8, Example 4.4]. By coherence [34, p.292] we have that $\overline{P}(X|\mathbf{B}) = X$ so that $\overline{P}(X) = \overline{P}(\overline{P}(X|\mathbf{B}))$.

Moreover $\inf_{B\in\mathbf{B}}\overline{P}(X|B) = \inf_{\omega\in\Omega}\overline{P}(X)$ and $\sup_{B\in\mathbf{B}}\overline{P}(X|B) = \sup_{\omega\in\Omega}\overline{P}(X)$.

Since $\overline{P}$ is a coherent upper probability we have that

$$\inf_{\omega\in\Omega}\overline{P}(X) \leq \overline{P}(X) \leq \sup_{\omega\in\Omega}\overline{P}(X)$$

and then the conglomerability principle is satisfied. □

.

Remark 9.1. Let Ω be a set with positive and finite Hausdorff outer measure in its Hausdorff dimension s. If S is a σ-field properly contained in $\wp(\Omega)$ and P is linear then h^s is σ-additive. In fact if P is linear then h^s is additive, moreover every Hausdorff outer measure is continuous from below and every additive measure defined on a σ-field is continuous from below if and only if it is σ-additive. Since h^s is σ-additive there exists at most a countable partition $\mathbf{B}$ such that each $B \in \mathbf{B}$ has positive and finite Hausdorff s-dimensional outer measure. Then for every S-measurable random variable X the disintegration property is satisfied for every countable partition $\mathbf{B}$ since the following equalities hold:

$$P(P(X|\mathbf{B})) = \sum_{B\in\mathbf{B}}\left(\frac{1}{h^s(B)}\int_B X dh^s\right)\frac{h^s(B)}{h^s(\Omega)} =$$

$$\frac{1}{h^s(\Omega)}\sum_{B\in\mathbf{B}}\int_B X dh^s = \frac{1}{h^s(\Omega)}\int_\Omega X dh^s = P(X)$$

9.7 Conclusions

The new model of upper conditional previsions proposed in this paper is based on the notion of Hausdorff dimensional outer measures defined in a metric space. An open problem is to find which properties of the given upper conditional previsions do not depend on the metric. For this aim is important to note that Hausdorff outer measures are defined in terms of diameters and covering properties of the conditioning event. Thus the Hausdorff outer measure of a set in its Hausdorff dimension is an intrinsic property of that set, as a set on which a metric function is defined. If a set B is removed from a metric space and re-embedded in another metric space such that the two metrics are equal on B this will not change the Hausdorff outer measure of B in its Hausdorff dimension.

Acknowledgements I wish to thank two referees and the Editors for their useful comments.

References

1. Artzner, P., Delbaen, F., Eber, J., Heath, D.: Coherent measures of risk. Math. Finance **3**, 203–228 (1999)
2. Bassanezi, R., Greco, G.: Sull'additivita' dell' integrale. Rend. Sem. Mat. Univ. Padova **72**, 249–275 (1984). (In Italian)

3. Billingsley, P.: Probability and measure. Chapman and Hall (1991)
4. Cabrelli, C., Hare, K.E., Molter, U.M.: Classifying Cantor sets by their fractal dimension. Proceedings of the American Mathematical Society **139**(11), 3965–3974 (2010)
5. Cabrelli, C., Mendivil, F., Molter, U.M., Shonkwiler, R.: On the Hausdorff h-measure of Cantor sets. Pacific Journal of Mathematics **217**(1), 45–60 (2004)
6. de Cooman, G., Troffaes, M., Miranda, E.: n-monotone exact functionals. Journal of Mathematical Analysis and Applications **347**, 133–146 (2008)
7. Croydon, D.: Hausdorff measure of arcs and Brownian motion on Brownian spatial trees. Annals of Probability **37**(3), 946–978 (2009)
8. Denneberg, D.: Non-additive measure and integral. Kluwer Academic Publishers (1994)
9. Doria, S.: Probabilistic independence with respect to upper and lower conditional probabilities assigned by Hausdorff outer and inner measures. International Journal of Approximate Reasoning **46**, 617–635 (2007)
10. Doria, S.: Characterization of a coherent upper conditional prevision as the Choquet integral with respect to its associated Hausdorff outer measure. Submitted to Annals of Operations Research (2010)
11. Doria, S.: Coherent upper conditional previsions and their integral representation with respect to Hausdorff outer measures. Advances in Intelligent and Soft Computing pp. 209–216 (2010)
12. Doria, S.: Stochastic independence with respect to upper and lower conditional probabilities assigned by Hausdorff outer and inner measures. Stochastic Control **46**, 87–101 (2010)
13. Dubins, L.: Finitely additive conditional probabilities, conglomerability and disintegrations. Annals of Probability **3**, 89–99 (1975)
14. Falconer, K.: The geometry of fractals sets. Cambridge University Press (1986)
15. de Finetti, B.: Teoria della Probabilita'. Einaudi (1970). (In Italian)
16. de Finetti, B.: Induction and Statistics. Wiley (1972)
17. Greco, G.: Sur la mesurabilité d'une fonction numérique par rapport à une famille d'ensembles. Rend. Sem. Mat. Univ. Padova **65**, 21–42 (1981). (In French)
18. Greco, G.: Sulla rappresentazione di funzionali mediante integrali. Rend. Sem. Mat. Univ. Padova pp. 21–42 (1982). (In Italian)
19. Kadane, J.B., Schervish, M., Seidenfeld, T.: Statistical implications of finitely additive probability. Bayesian Inference and Decision Techniques With Applications pp. 59–76 (1986)
20. Koch, G.: La matematica del probabile. Aracne Editrice (1997). (In Italian)
21. Maaß, S.: Exact functionals and their core. Statistical papers **45**(1), 75–93 (2002)
22. Miranda, E., Zaffalon, M.: Conditional models: coherence and inference trough sequences of joint mass functions. Journal of Statistical Planning and Inference **140(7)**, 1805–1833 (2009)
23. Narukawa, Y., Murofushi, T., Sugeno, M.: Regular fuzzy measure and representation of comonotonically additive functionals. Fuzzy Sets and Systems **112**, 177–186 (2000)
24. Perkins, E.: The exact Hausdorff measure of the level sets of Brownian motion. Probability theory and related fields (1981)
25. Regazzini, E.: Finitely additive conditional probabilities. Rend. Sem. Mat. Fis. **58**(3), 373–388 (1985)
26. Regazzini, E.: De Finetti's coherence and statistical inference. The Annals of Statistics **15**(2), 845–864 (1987)
27. Rogers, C.: Hausdorff measures. Cambridge University Press (1970)
28. Schervish, M., Seidenfeld, T., Kadane, J.: The extent of non-conglomerability of finitely additive probabilities. Z. Warsch.Verw.Gebiete **66**, 205–226 (1984)
29. Schmeidler, D.: Cores of exact games. I.J.Math.Anal.Appl. **40**, 214–225 (1972)
30. Schmeidler, D.: Integral representation without additivity. Proceedindgs of the American Mathematical Society **97**, 225–261 (1986)
31. Scozzafava, R.: Probabilita' σ-additive e non. Bollettino U.M.I. **57**(1-A), 1–33 (1986). (In Italian)
32. Vicig, P., Zaffalon, M., Cozman, F.: Notes on "Notes on conditional previsions". International Journal of Approximate Reasoning **44**, 358–365 (2007)
33. Walley, P.: Coherent lower (and upper) probabilities. Statistics Research Report, University of Warwick (1981)

34. Walley, P.: Statistical Reasoning with Imprecise Probabilities. Chapman and Hall (1991)
35. Williams, P.: Notes on conditional previsions. International Journal of Approximate Reasoning **44**, 366–383 (2007)

Part II
Applications: Uncertainties in Engineering

The second part of this book is concerned with real-life application scenarios from various areas including but not limited to mechatronics, robotics, and biomedical engineering. In Chapter 10, Marco Kletting, Michel Kieffer, and Eric Walter give a detailed comparison of two different approaches for guaranteed state and parameter estimation. Mehrdad Moshir presents a case study on the quantification of spacecraft failures in uncertain environments in Chapter 11. Chapter 12 by Denis Efimov, Tarek Raïssi, and Ali Zolghadri focuses on robust state and parameter estimation techniques for nonlinear systems. Nonlinear adaptive control approaches for bioprocesses are presented by Neli Dimitrova and Mikhail Krastanov in Chapter 13. Ekaterina Auer, Haider Albassam, Andrés Kecskeméthy, and Wolfram Luther discuss the verified analysis of a mathematical model for stance stabilization in Chapter 14. Chapter 15 by Vasily V. Saurin, Georgy V. Kostin, Andreas Rauh, and Harald Aschemann deals with the adaptive control of heat transfer problems with uncertainties. Finally, Harald Aschemann, Dominik Schindele, and Jöran Ritzke discuss state and disturbance estimation for the control of high-speed rack feeders in the last chapter of this book.

Chapter 10
Two Approaches for Guaranteed State Estimation of Nonlinear Continuous-Time Models

Marco Kletting, Michel Kieffer (✉), and Eric Walter

Abstract This paper deals with the estimation of the state vector of a nonlinear continuous-time state-space model, such as those frequently encountered in the context of knowledge-based modeling. Unknown and possibly time-varying parameters may be included in an extended state vector to deal with the simultaneous estimation of state and parameters. Observations depending on the (possibly extended) state are assumed to take place at discrete measurement times. Given bounds on the size of the additive measurement errors, guaranteed estimation should then provide bounds on the possible values of the state at any given time. Two recently developed approaches are presented and their performance is compared on a simple test case.

10.1 Introduction

When building knowledge-based models, for instance models based on the laws of physics, one frequently ends up with a continuous-time state-space model, which may depend on a possibly time-varying vector of parameters $\mathbf{p} \in \mathbb{R}^{n_p}$:

Marco Kletting
Multi-Function Airborne Radars (OPES22), Cassidian Electronics,
Woerthstr. 85, 89077 Ulm, Germany
e-mail: marco.kletting@cassidian.com

Michel Kieffer
Laboratoire des Signaux et Systèmes - CNRS - SUPELEC - Univ Paris-Sud, 3 rue Joliot-Curie,
F-91192 Gif-sur-Yvette cedex, on leave at LTCI - CNRS - Telecom ParisTech, 46 rue Barault,
F-75013 Paris, France
e-mail: michel.kieffer@lss.supelec.fr

Eric Walter
Laboratoire des Signaux et Systèmes - CNRS - SUPELEC - Univ Paris-Sud, 3 rue Joliot-Curie,
F-91192 Gif-sur-Yvette cedex, France
e-mail: eric.walter@lss.supelec.fr

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_10,

$$\dot{\mathbf{x}}(t) = \mathbf{f}_x(\mathbf{x}(t), \mathbf{p}(t)),$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is the state vector. Provided that an equation is specified for the dynamics of $\mathbf{p}$, such as

$$\dot{\mathbf{p}}(t) = \Delta\mathbf{p} \quad \text{with} \quad \Delta\mathbf{p} \in [\Delta\underline{\mathbf{p}}, \Delta\overline{\mathbf{p}}],$$

one can concatenate the state and parameter vectors into an extended state vector $\mathbf{z}(t) = [\mathbf{x}(t)^T, \mathbf{p}(t)^T]^T$. Then

$$\dot{\mathbf{z}}(t) = \mathbf{f}(\mathbf{z}(t)), \tag{10.1}$$

where

$$\mathbf{f} = \begin{bmatrix} \mathbf{f}_x(\mathbf{x}(t), \mathbf{p}(t)) \\ \Delta\mathbf{p} \end{bmatrix}$$

with $\mathbf{f} : D \mapsto \mathbb{R}^n$, $D \subset \mathbb{R}^n = \mathbb{R}^{n_x} \times \mathbb{R}^{n_p}$. Any time-invariant parameter p_i should satisfy $\Delta p_i = 0$.

This paper is devoted to the estimation of the state of such a model from discrete-time measurements. Measurement times $t_1 < t_2 < \cdots < t_{k_{max}}$ may not be regularly spaced. The step from t_k to t_{k+1}, with size $h_k = t_{k+1} - t_k$, is referred to as the $(k+1)$-st step. The vector $\mathbf{y}(t_k) \in \mathbb{R}^m$ of the values measured at time t_k is assumed to satisfy

$$\mathbf{y}(t_k) = \check{\mathbf{y}}(t_k) + \boldsymbol{\delta}(t_k), \tag{10.2}$$

where $\boldsymbol{\delta}(t_k)$ is an additive measurement error vector and

$$\check{\mathbf{y}}(t_k) = \mathbf{h}(\mathbf{z}(t_k), t_k) \tag{10.3}$$

is the result that would have been obtained from ideal measurements. Usually, $n < m$. The absolute value of the measurement error is assumed to be bounded, with a known upper bound $\overline{\boldsymbol{\delta}}$, which implies that $\boldsymbol{\delta}(t_k) \in [-\overline{\boldsymbol{\delta}}; \overline{\boldsymbol{\delta}}]$ for all k. As a result, we have

$$\check{\mathbf{y}}(t_k) \in [\mathbf{y}(t_k)] = [\mathbf{y}(t_k) - \overline{\boldsymbol{\delta}}, \mathbf{y}(t_k) + \overline{\boldsymbol{\delta}}] \ .$$

If there are uncertain parameters in the measurement equation, then they can be incorporated in the extended state vector, just as we have done with the uncertain parameters in the state equation. In what follows, we shall therefore only be concerned with the computation of guaranteed estimates of the (extended) state vector, which contain guaranteed estimates of the parameters of the state and observation equations, if any. This is why the extended state will simply be called state in what follows, unless we need to distinguish the parameters from the state proper.

Guaranteed nonlinear state estimation in this context of bounded-measurement errors have been addressed by a number of authors, see *e.g.* [2, 3, 7, 10, 15, 16, 22, 23, 26, 27, 30, 32]. All these approaches enclose the set of all state values consistent with the model, and the measurements and noise bounds. They differ first by the wrappers used to perform enclosure. Ellipsoids are used in [2, 3, 16, 30], zonotopes in [15], boxes in [7, 10, 22, 23, 26, 27], and union of boxes in [32]. A second important difference is in the hypotheses about the measurements. In [22, 23, 27], continuous-time measurements are assumed to be available. This rather unrealistic assumption allows nice convergence properties of the estimators to be obtained. Here, discrete-time measurements will be considered, which appears more realistic, but makes convergence analysis much more difficult, especially for nonlinear systems. Techniques using boxes or union of boxes as wrappers usually rely on interval analysis and guaranteed integration of ODEs.

We shall assume in this chapter that bounds are available on the possible value of the initial state, and that estimates of the present state are to be produced based on the past measurements only. As for most state estimators including the celebrated Kalman filter, the $(k+1)$-st step of the procedure will then consist of two steps: a *prediction step* that predicts the evolution of the state between two instants of time at which measurements are obtained, and a *correction step* during which the newly acquired data are taken into account to reduce the uncertainty in the result of the prediction step. In the context of guaranteed estimation, the prediction step involves guaranteed integration under consideration of all uncertainties, and the correction step must eliminate any part of the predicted set that can be proved to be inconsistent with the new measurements given the bounds on the measurement errors. For the sake of simplicity, no state perturbation has been considered here. On how state perturbations may be considered, see, *e.g.*, [12].

Section 10.2 overviews bounded-error state estimation in an idealized context. Section 10.3 considers two approaches that rely on interval analysis and try to counteract the pessimism of guaranteed integration when the initial conditions are uncertain, as is the case in an estimation context. The first of these approaches is based on Müller's theorem and presented in Section 10.3.1, while the second, based on the use of high-order Taylor models is described in Section 10.3.2. Various ways to implement correction steps are then described in Section 10.4. The resulting algorithms are compared in Section 10.5 on a simple test-case.

10.2 Idealized State Estimation

Let $\mathbb{Z}(t)$ be the set of all state values that are consistent with the information available up to time t. An idealized bounded-error counterpart of the Kalman filter for nonlinear discrete-time systems, alternating prediction and correction steps [8], may be considered to build $\mathbb{Z}(t)$ at each t.

If $\mathbf{z}(t_k)$ at t_k is only known to belong to $\mathbb{Z}(t_k)$, the *set* of solutions at time $t > t_k$ of (10.1) is

$$\mathbf{z}(t;t_k,\mathbb{Z}(t_k)) = \{\mathbf{z}(t;t_k,\mathbf{z}(t_k)) \mid \mathbf{z}(t_k) \in \mathbb{Z}(t_k)\}.$$

For the $(k+1)$-st prediction step, one has thus to compute the *predicted* set

$$\mathbb{Z}^+(t_{k+1}) = \mathbf{z}(t_{k+1};t_k,\mathbb{Z}(t_k)).$$

For the $(k+1)$-st correction step, one has to take into account the measurement available at time t_{k+1} to update $\mathbb{Z}^+(t_{k+1})$ and obtain

$$\mathbb{Z}(t_{k+1}) = \left\{ \mathbf{z} \in \mathbb{Z}^{+}(t_{k+1}) \,|\, \mathbf{h}(\mathbf{z}, t_{k+1}) \in [\mathbf{y}(t_{k+1}) - \overline{\boldsymbol{\delta}}, \mathbf{y}(t_{k+1}) + \overline{\boldsymbol{\delta}}] \right\}, \tag{10.4}$$

thus $\mathbb{Z}(t_{k+1}) \subseteq \mathbb{Z}^{+}(t_{k+1})$.

Provided that all hypotheses on the state equation and the measurement noise are satisfied, $\mathbb{Z}(t_{k+1})$ does contain $\mathbf{z}(t_{k+1})$. The main difficulty in this idealized algorithm comes from the fact that $\mathbb{Z}(t_k)$ may have a quite complex shape. Outer-approximations of the sets $\mathbb{Z}$ and $\mathbb{Z}^{+}$ have thus to be evaluated. These outer-approximations may consist of interval vectors (boxes), union of non-overlapping boxes (subpavings), or may be described using Taylor models. Implementable counterparts of the idealized prediction and correction steps are now described.

10.3 Prediction Step

A naive approach to obtain an outer-approximation for $\mathbf{z}(t_k; t_0, [\mathbf{z}(0)])$, $k = 1 \dots k_{max}$, would be to use one of the guaranteed ODE solvers based on interval analysis `AWA` [18], `VNODE` [25], `COSY IV` [6], `VSPODE` [17], or `ValEncIA-IVP` [28,29]. The main difficulty is to obtain accurate enclosures for the solutions, when there are uncertain parameters, bounded state perturbation, or uncertain initial conditions.

One may enclose the solutions of (10.1) with *uncertain* initial conditions between a pair of coupled system of ODEs with *deterministic* initial conditions using Müller's theorem [24]. Other types of uncertainty may be taken into account as well, such as unknown but bounded inputs. Any guaranteed tool for solving ODEs may then be used to solve this system, see Section 10.3.1.

When only the initial conditions are undetermined, but known to belong to some box, guaranteed ODE solvers such as `COSY IV`, `VSPODE`, or `ValEncIA-IVP` perform quite well, since they are evaluating a Taylor development of the solution with interval remainder, this developments being made also with respect to the initial condition, see Section 10.3.2.

10.3.1 Using Müller's Theorem

Here, the solution is obtained by bounding the solutions of dynamical systems with *uncertain* parameters or initial conditions using *deterministic* dynamical systems. This approach has been previously presented in [11,31] in the context of cooperative dynamical models, *i.e.*, models such as (10.1) for which the off-diagonal terms of the Jacobian matrix of $\mathbf{f}$ are positive. These results were inspired by the interval observer proposed in [5]. Müller's theorems [24], which have recently been used in the context of guaranteed simulation [4], make it possible to bound the solutions of more general dynamical models, see also [13].

10.3.1.1 Müller's Theorem

The following theorem [24] may be directly applied to bound the solutions of dynamical models such as (10.1) in the presence of uncertain initial conditions $\mathbf{z}(t_k) \in [\mathbf{z}(t_k)]$, where $[\mathbf{z}(t_k)]$ is a box in state space.

Theorem 10.1. *Assume that* $\mathbf{f}(\mathbf{z}(t))$ *in* (10.1) *is continuous on*

$$\mathbb{T} : \begin{cases} \boldsymbol{\omega}(t) \leqslant \mathbf{x} \leqslant \boldsymbol{\Omega}(t) \\ t_k \leqslant t \leqslant t_{k+1} \end{cases}$$

where $\omega_i(t)$ *and* $\Omega_i(t)$, $i = 1 \ldots n_x$, *are continuous on* $[t_k, t_{k+1}]$ *and satisfy*

1. $\boldsymbol{\omega}(t_k) = \underline{\mathbf{z}}(t_k)$ *and* $\boldsymbol{\Omega}(t_k) = \overline{\mathbf{z}}(t_k)$,
2. *the left derivatives* $D^-\omega_i(t)$ *and* $D^-\Omega_i(t)$ *and right derivatives* $D^+\omega_i(t)$ *and* $D^+\Omega_i(t)$ *of* $\omega_i(t)$ *and* $\Omega_i(t)$ *satisfy, for* $i = 1 \ldots n$ *and all* $t \in [t_k, t_{k+1}]$,

$$D^{\pm}\omega_i(t) \leqslant \min_{\underline{\mathbb{T}}_i(t)} f_i(\mathbf{z}) \qquad \textit{and} \qquad D^{\pm}\Omega_i(t) \geqslant \max_{\overline{\mathbb{T}}_i(t)} f_i(\mathbf{z}),$$

where $\underline{\mathbb{T}}_i(t)$ *is the subsets of D defined by*

$$\underline{\mathbb{T}}_i(\tau) : \begin{cases} z_i = \omega_i(\tau), \\ \omega_j(\tau) \leqslant z_j \leqslant \Omega_j(\tau),\ j \neq i, \\ t = \tau, \end{cases}$$

and $\overline{\mathbb{T}}_i(t)$ *is the subset of D defined by*

$$\overline{\mathbb{T}}_i(\tau) : \begin{cases} z_i = \Omega_i(\tau) \\ \omega_j(\tau) \leqslant z_j \leqslant \Omega_j(\tau),\ j \neq i, \\ t = \tau. \end{cases}$$

Then, for any given $\mathbf{z}(t_k) \in [\mathbf{z}(t_k)]$, *a solution to* (10.1) *exists, such that*

$$\boldsymbol{\omega}(t) \leqslant \mathbf{z}(t) \leqslant \boldsymbol{\Omega}(t) \quad \forall t \in [t_k, t_{k+1}],$$

and this solution is equal to $\mathbf{z}(t_k)$ *at* $t = t_k$. *Moreover, if for any* $t \in [t_k, t_{k+1}]$, $\mathbf{f}(\mathbf{z}, t)$ *is Lipschitz with respect to* $\mathbf{z}$, *then for any given* $\mathbf{z}(t_k) \in [\mathbf{z}(t_k)]$ *this solution is unique.* ♢

The main idea of this theorem is to bracket the solutions of (10.1) between the solution of two deterministic ODEs. The initial conditions for these ODEs are given by 1. and the conditions that have to be satisfied by each solution are given by 2., see Section 10.5 for an example.

10.3.1.2 Prediction Step Using Müller's Theorem

Müller's theorem allows the evaluation of lower and upper bounds for the solution of (10.1) provided that two functions $\boldsymbol{\omega}(t)$ and $\boldsymbol{\Omega}(t)$ are available. The interval function $[\boldsymbol{\Phi}](t) = [\boldsymbol{\omega}(t), \boldsymbol{\Omega}(t)]$ can then be seen as an *inclusion function* for all solutions $\mathbf{z}(t)$ of the state equation in (10.1).

$[\boldsymbol{\Phi}](t)$ provides a *box* containing the state at each time instant. More accurate descriptions of the predicted set may be obtained by *splitting* the box corresponding to the initial conditions into non-overlapping subboxes, and to apply Müller's theorem on each of the resulting subboxes. A list of overlapping boxes is obtained, which may be merged into a subpaving using the `ImageSp` algorithm [8]. This subpaving may then be used by any implementable correction step described in Section 10.4, or used to perform a new prediction until a measurement is available. The accuracy of the description of $\mathbb{Z}^+(t_k)$ at time t_k depends on some precision parameter ε, which determines the size of the boxes obtained after splitting the initial box or subpaving used by the `ImageSp` algorithm.

The construction of $\boldsymbol{\omega}(t)$ and $\boldsymbol{\Omega}(t)$ is usually easy on a case-by-case basis, as illustrated in Section 10.5.1. For more complex systems, hybrid automata may be put at work to build these functions, as detailed in [21].

An inclusion function for $\mathbf{h}$ may similarly be obtained using the sensitivity functions of the output with respect to the state, see Section 10.4.2.3.

10.3.2 Verified Integration Based on Taylor Models

Verified integration techniques such as `VNODE` [25] are based on a Taylor series expansion in time. `COSY-VI` [1] performs, in addition to this expansion in time, an expansion with respect to the initial state vector, denoted by $\mathfrak{z}$ in what follows. The box to which $\mathfrak{z}$ is assumed to belong is given by $[\mathfrak{z}]$. The expansion point with respect to the initial state vector $\mathfrak{z}$ is some $\hat{\mathfrak{z}} \in [\mathfrak{z}]$, and the expansion point with respect to time is t_k. The flow of the differential equation in a given time interval $[t_k, t_{k+1}]$ is enclosed by a n-dimensional Taylor model

$$\begin{aligned} &\mathbf{T}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k) := \mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k) + \mathbf{I}_{\rho,k+1}, \\ &\text{with} \quad \mathfrak{z} \in [\mathfrak{z}] \quad \text{and} \quad t \in [t_k, t_{k+1}] \ , \end{aligned}$$

where $\mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)$ is the multivariate polynomial part of order ρ and $\mathbf{I}_{\rho,k+1}$ is the remainder box. The i-th entry of the n-dimensional vector $\mathbf{T}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)$ is denoted by $T_{\rho,i}(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)$. The Taylor model at $t = t_{k+1}$ is written as

$$\mathbf{T}_{\rho,k+1}(\mathfrak{z}-\hat{\mathfrak{z}}) := \mathbf{P}_{\rho,k+1}(\mathfrak{z}-\hat{\mathfrak{z}}) + \mathbf{I}_{\rho,k+1} \ .$$

The i-th entry of $\mathbf{T}_{\rho,k+1}(\mathfrak{z}-\hat{\mathfrak{z}})$ is given by $T_{\rho,i,k+1}(\mathfrak{z}-\hat{\mathfrak{z}})$.

Verified integration methods that use a single box or a single parallelepiped for the state enclosure may suffer from large overestimation, especially for nonlinear systems. The flow representation by Taylor models makes it possible to obtain tight enclosures of non-convex sets and leads to less overestimation.

The integral form of the differential equation (10.1) is given by

$$\mathscr{O}(\mathbf{z}(t)) := \mathbf{z}(t_k) + \int_{t_k}^{t} \mathbf{f}(\mathbf{z}(t'),t')dt'$$

Applying $\mathscr{O}$ to a Taylor model for the integration in the time-interval $[t_k, t_{k+1}]$ yields

$$\mathscr{O}(\mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)+\mathbf{I}_{\rho,k+1}) = \mathbf{z}(t_k) + \int_{t_k}^{t} \mathbf{f}(\mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t'-t_k)+\mathbf{I}_{\rho,k+1})dt',$$

where $\mathbf{z}(t_k)$ is represented by its Taylor model enclosure at $t = t_k$.

$$\mathbf{T}_{\rho,k} = \mathbf{P}_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}}) + \mathbf{I}_{\rho,k} \quad .$$

This leads to

$$\begin{aligned}&\mathscr{O}(\mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)+\mathbf{I}_{\rho,k+1})\\ &= \mathbf{P}_{\rho,k}(\mathfrak{z}-\hat{\mathfrak{z}}) + \mathbf{I}_{\rho,k} + \int_{t_k}^{t} \mathbf{f}(\mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)+\mathbf{I}_{\rho,k+1})dt'.\end{aligned}$$

The goal for the integration from t_k to t_{k+1} consists in determining a Taylor model $\mathbf{T}_\rho\,(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)$ such that

$$\mathscr{O}(\mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)+\mathbf{I}_{\rho,k+1}) \subset \mathbf{P}_\rho(\mathfrak{z}-\hat{\mathfrak{z}}, t-t_k)+\mathbf{I}_{\rho,k+1}$$

$\forall \mathfrak{z} \in [\mathfrak{z}]$ and $\forall t \in [t_k, t_{k+1}]$. The polynomial part and the interval remainder are determined in separate steps. A detailed description of these steps is given in [1, 14].

For numerical and implementation reasons it is advantageous to have the unit box $[-1;1]^n$ as the domain box in each integration step [20]. Thus the initial enclosure $[\mathbf{z}(0)]$ of the extended state vector $\mathbf{z}(t)$ is expressed as a Taylor model according to

$$\begin{aligned}[\mathbf{z}(0)] = \mathbf{T}(\mathfrak{z}) &= \mathbf{c} + \mathbf{D}\mathfrak{z}\\ &\text{with} \quad \mathfrak{z}_i \in [-1;1], \;\; i = 1 \ldots n,\end{aligned}$$

where $\mathbf{c}$ is the midpoint of $[\mathbf{z}(0)]$ and $\mathbf{D}$ is a diagonal matrix with $d_{i,i} = \mathrm{rad}([\mathbf{z}(0)])$.

The expansion in initial state reduces the overestimation that may occur during integration. To limit the long-term growth of the remainder error and to reduce overestimation the following strategies can be applied:

- Shrink Wrapping: the interval remainder is absorbed in the polynomial part [20].
- Preconditioning: the ODE solution is studied in a more suitable coordinate system [19].
- Splitting: the domain box $[\mathfrak{z}]$ is split into subboxes and the enclosure of $\mathbf{z}(t)$ is given by a list of Taylor models [14]. This is described in the following section.

10.3.2.1 Splitting the Domain Box

As when using Müller's theorem, splitting of the domain box into subboxes [14] may be useful to reduce overestimation. The state vector $\mathbf{z}_{k+1}$ at $t = t_{k+1}$ is then enclosed by a list $\mathscr{T}_{k+1}$ of Taylor models

$$\mathbf{z}_{k+1} \in \mathscr{T}_{k+1} = \left\{\mathbf{T}^{(1)}_{\rho,k+1}(\mathfrak{z}), \mathbf{T}^{(2)}_{\rho,k+1}(\mathfrak{z}) \dots \mathbf{T}^{(L_{k+1})}_{\rho,k+1}(\mathfrak{z})\right\}$$
$$\text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1 \dots n \quad \text{and} \quad L_{k+1} \leq L_{max}.$$

where L_{max} is the maximum allowed number of Taylor models. Consider a Taylor model $\mathbf{T}_{\rho,k+1}(\mathfrak{z})$ with the domain box $[\mathfrak{z}]$, $\mathfrak{z} \in [\mathfrak{z}]$. The domain box of this Taylor model is split into subboxes $[\mathfrak{z}^{(l)}]$, $l = 1 \dots L$,

$$\bigcup_{l=1}^{L} [\mathfrak{z}^{(l)}] = [\mathfrak{z}] \ .$$

To obtain again the unit box as a domain box, $[\mathfrak{z}^{(l)}]$ is expressed as a Taylor model according to

$$[\mathfrak{z}^{(l)}] = \widetilde{\mathbf{T}}^{(l)}(\mathfrak{z}) = \mathbf{c}^{(l)} + \mathbf{D}^{(l)}\mathfrak{z}$$
$$\text{with} \quad \mathfrak{z}_i \in [-1;1] \ , i = 1 \dots n \ ,$$

where $\mathbf{c}^{(l)}$ is the midpoint of $[\mathfrak{z}^{(l)}]$ and $\mathbf{D}^{(l)}$ is a diagonal matrix with $d^{(l)}_{i,i} = \mathrm{rad}([\mathfrak{z}^{(l)}_i])$. The components of the original initial state vector $\mathfrak{z}$ of $\mathbf{T}_{\rho,k+1}(\mathfrak{z})$ are replaced by the components of $\widetilde{\mathbf{T}}^{(l)}(\mathfrak{z})$ by substituting $\widetilde{T}^{(l)}_i(\mathfrak{z})$ for $\mathfrak{z}_i$, which results in a modified Taylor model

$$\mathbf{T}^{(l)}_{\rho,k+1}(\mathfrak{z}) = \mathbf{T}_{\rho,k+1}(\widetilde{\mathbf{T}}^{(l)}(\mathfrak{z}))$$

for each subbox $[\mathfrak{z}^{(l)}]$.

To determine the component in which the domain box has to be split, splitting criteria have to be evaluated for the considered Taylor model $\mathbf{T}_{\rho,k+1}(\mathfrak{z})$. Splitting is carried out perpendicularly to the direction which has been calculated by the splitting criteria. Approaches to determine the splitting direction are described in [14].

If several Taylor models are already present, the most appropriate Taylor model for the splitting has to be selected. This is done by calculating the interval enclosure of each Taylor model and the corresponding pseudo volume of the resulting box. The pseudo volume of a n-dimensional interval vector is calculated by the multiplication of the interval diameters of all its components. The Taylor model with the largest pseudo volume is selected. Alternatively, the Taylor model with the largest interval remainder could be selected.

How splitting of the domain box is combined with preconditioning is described in detail in [14].

10.3.2.2 Prediction Step Using Taylor Models

At time t_k, the extended state vector is enclosed by a list $\mathscr{T}_k$ of Taylor models

$$\mathbf{z}_k \in \mathscr{T}_k = \left\{ \mathbf{T}^{(1)}_{\rho,k}(\mathfrak{z}), \mathbf{T}^{(2)}_{\rho,k}(\mathfrak{z}) \ldots \mathbf{T}^{(L_k)}_{\rho,k}(\mathfrak{z}) \right\}$$
$$\text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1 \ldots n \quad \text{and} \quad L_k \leq L_{max}.$$

First a Taylor model is selected for splitting, then a splitting criterion is evaluated, and the Taylor model is split by splitting the domain box in subboxes. Taylor models are split until a pre-specified number of Taylor models or number of splittings is reached. Next, for each Taylor model a verified integration is performed.

The resulting enclosure of the extended state vector at time t_{k+1} (after the prediction step) is then given by

$$\mathscr{T}^{pr}_{k+1} = \left\{ \mathbf{T}^{(pr,1)}_{\rho,k+1}(\mathfrak{z}), \mathbf{T}^{(pr,2)}_{\rho,k+1}(\mathfrak{z}) \ldots \mathbf{T}^{(pr,L^{pr}_{k+1})}_{\rho,k+1}(\mathfrak{z}) \right\}$$
$$\text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1 \ldots n \quad \text{and} \quad L^{pr}_{k+1} \leq L_{max}.$$

The prediction step is repeated until measurements become available. The result of the last prediction step before measurements become available is used as an initial enclosure of the next correction step.

10.4 Correction Step

Assume that the prediction step at time t_{k+1} has produced a set $\mathbb{Z}^+(t_{k+1})$ that contains $\mathbf{z}(t_{k+1})$. This set may consist of a single box, a list of potentially overlapping boxes, or may be a subpaving.

The measurement vector $\mathbf{y}(t_{k+1})$ obtained at time t_{k+1} has now to be taken into account. Several practical implementations of the idealized correction step (10.4) are now presented.

10.4.1 Using Set Inversion Via Interval Analysis

The aim of the Set Inverter Via Interval Analysis (SIVIA) algorithm [9] is to eliminate parts of $\mathbb{Z}^+(t_{k+1})$ that can be proved to be inconsistent with the measurements, the measurement equation and the bounds on the measurement noise.

Consider a box $[\mathbf{z}] \subset \mathbb{Z}^+(t_{k+1})$. If $\mathbf{h}([\mathbf{z}], t_{k+1}) \subset [\mathbf{y}(t_{k+1}) - \overline{\boldsymbol{\delta}}, \mathbf{y}(t_{k+1}) + \overline{\boldsymbol{\delta}}]$, then all $\mathbf{z} \in [\mathbf{z}]$ are consistent with the measurements, model, and noise bounds. Therefore, $[\mathbf{z}]$ has been proved to belong to $\mathbb{Z}(t_{k+1})$. If $\mathbf{h}([\mathbf{z}], t_{k+1}) \cap [\mathbf{y}(t_{k+1}) - \overline{\boldsymbol{\delta}}, \mathbf{y}(t_{k+1}) + \overline{\boldsymbol{\delta}}] =$

$\emptyset$, then no $\mathbf{z} \in [\mathbf{z}]$ is consistent with the measurement, model and noise bounds. Thus, $[\mathbf{z}]$ has an empty intersection with $\mathbb{Z}(t_{k+1})$ and can be rejected. If none of the previous conditions is satisfied, $[\mathbf{z}]$ is said undetermined, as parts of $[\mathbf{z}]$ may belong to $\mathbb{Z}(t_{k+1})$.

The `SIVIA` algorithms iteratively performs selection, elimination, or bisection of boxes, starting from a large initial search box, list of boxes or subpaving. Undetermined boxes are bisected until their width is smaller than some precision parameter ε, which helps to trade-off complexity and accuracy of representation of the solution set, $\mathbb{Z}(t_{k+1})$ here. The solution provided by `SIVIA` is a subpaving, consisting of inner boxes and undetermined boxes deemed too small to be further bisected. This subpaving may be fed to the next prediction step. See [8] for more details and implementation issues.

10.4.2 Using Contractors

Bounded-error measurements translate into vector inequality constraint $\mathbf{k}(\mathbf{z}) \geqslant \mathbf{0}$, to be understood componentwise. A *contractor* $C_{\mathbf{k}}$ for $\mathbf{z}$ is an algorithm to compute a box $C_{\mathbf{k}}([\mathbf{z}])$ such that

$$\begin{cases} C_{\mathbf{k}}([\mathbf{z}]) \subset [\mathbf{z}], \\ \{\mathbf{z} \in [\mathbf{z}] \,|\mathbf{k}(\mathbf{z}) \geqslant \mathbf{0}\} \subset C_{\mathbf{k}}([\mathbf{z}]). \end{cases} \tag{10.5}$$

The first relation in (10.5) ensures that $[\mathbf{z}]$ is contracted, while the second guarantees that no value of $\mathbf{z}$ satisfying the constraints is lost. Contractors can be similarly defined in the case of equality constraints.

10.4.2.1 Improving the State Estimate

Given the measurement equations (10.2), (10.3) and the fact that $\delta_k \in \left[-\overline{\delta}, \overline{\delta}\right]$, two constraints may be obtained that have to be satisfied by the state vector at time t_k, namely

$$\mathbf{k}_1(\mathbf{z}) = \mathbf{y}(t_k) - \mathbf{h}(\mathbf{z}, t_k) + \overline{\delta} \geqslant \mathbf{0}$$

and

$$\mathbf{k}_2(\mathbf{z}) = -\mathbf{y}(t_k) + \mathbf{h}(\mathbf{z}, t_k) + \overline{\delta} \geqslant \mathbf{0}.$$

Various types of contractors may be considered [8], depending on the structure of $\mathbf{h}(\mathbf{z}, t_k)$. If $m = n$, the interval Newton or the Krawczyk contractors may be employed.

10.4.2.2 Improving the Initial Conditions Estimate

If one is interested in obtaining a better estimate of the initial conditions, one may write $\mathbf{h}(\mathbf{z},t_k)$ as a function of $\mathfrak{z}$. For that purpose, consider the i-th entry of $\mathbf{h}(\mathbf{z},t_k)$. We have

$$h_i([\mathbf{z}_{k+1}],t_k) \subset h_i([\mathbf{z}_{k+1}](\widehat{\mathfrak{z}}),t_k) + ([\mathfrak{z}]-\widehat{\mathfrak{z}})^{\mathrm{T}} \left.\frac{\partial h_i(\mathbf{z},t_k)}{\partial \mathfrak{z}}\right|_{[\mathbf{z}_{k+1}]}, \tag{10.6}$$

where $\widehat{\mathfrak{z}} \in [\mathfrak{z}]$ and $[\mathbf{z}_{k+1}](\widehat{\mathfrak{z}})$ is the box obtained when integrating the system with known initial conditions taken as $\widehat{\mathfrak{z}}$. In (10.6),

$$\frac{\partial h_i(\mathbf{z},t_k)}{\partial \mathfrak{z}} = \begin{pmatrix} \frac{\partial z_1}{\partial \mathfrak{z}_1} & \cdots & \frac{\partial z_n}{\partial \mathfrak{z}_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_1}{\partial \mathfrak{z}_n} & \cdots & \frac{\partial z_n}{\partial \mathfrak{z}_n} \end{pmatrix} \begin{pmatrix} \frac{\partial h_i}{\partial z_1} \\ \vdots \\ \frac{\partial h_i}{\partial z_n} \end{pmatrix}$$

where $\frac{\partial z_i}{\partial \mathfrak{z}_j}$ is the *sensitivity function* of the i-th state component with respect to the initial condition of the j-th state component.

Taking into account the measurement at time t_k,

$$h_i([\mathbf{z}_{k+1}](\widehat{\mathfrak{z}}),t_k) + ([\mathfrak{z}]-\widehat{\mathfrak{z}})^{\mathrm{T}} \left.\frac{\partial h_i(\mathbf{z},t_k)}{\partial \mathfrak{z}}\right|_{[\mathbf{z}_{k+1}]} \subset \left[y_i(t_k)-\overline{\delta}, y_i(t_k)+\overline{\delta}\right],\ i=1\ldots m.$$

Thus each $[\mathfrak{z}_i]$, $i=1\ldots n$, has to satisfy

$$\begin{aligned}[\mathfrak{z}_i] \subset \Big(& y(t_k) - \left[-\overline{\delta},\overline{\delta}\right] - h_i([\mathbf{z}_{k+1}](\widehat{\mathfrak{z}}),t_k) \\ & - \sum_{j\neq i} ([\mathfrak{z}_j]-\widehat{\mathfrak{z}}_j)\left[\frac{\partial h_i}{\partial \mathfrak{z}_j}\right]([\mathbf{z}_{k+1}])\Big) \Big/ \left[\frac{\partial h_i}{\partial \mathfrak{z}_i}\right]([\mathbf{z}_{k+1}]) + \widehat{\mathfrak{z}}_i,\end{aligned}$$

leading to a contracted box $[\mathfrak{z}]^{\text{new}} = C_i([\mathfrak{z}])$, the components of which are defined as

$$\begin{aligned}[\mathfrak{z}_i]^{\text{new}} = [\mathfrak{z}_i] \cap \Big(\Big(& y(t_k) - \left[-\overline{\delta},\overline{\delta}\right] - h_i([\mathbf{z}_{k+1}](\widehat{\mathfrak{z}}),t_k) \\ & - \sum_{j\neq i} ([\mathfrak{z}_j]-\widehat{\mathfrak{z}}_j)\left[\frac{\partial h_i}{\partial \mathfrak{z}_j}\right]([\mathbf{z}_{k+1}])\Big) \Big/ \left[\frac{\partial h_i}{\partial \mathfrak{z}_i}\right]([\mathbf{z}_{k+1}]) + \widehat{\mathfrak{z}}_i\Big)\end{aligned}$$

for $i=1\ldots n$.

This contractor requires the computation of an inclusion function for the sensitivity functions of each state component with respect to the initial condition; see the next section.

10.4.2.3 Sensitivity Functions

Denote the first-order sensitivity of z_j with respect to $\mathfrak{z}_k$ by s_{jk}

$$s_{jk}(\mathfrak{z},t) = \frac{\partial z_j}{\partial \mathfrak{z}_k}(\mathfrak{z},t).$$

Assume for simplicity that the model output is linear in the state and it is given by

$$\mathbf{h}(\mathbf{z}(t),t) = \mathbf{C}\mathbf{z}(t),$$

where $\mathbf{C} = (\mathbf{c}_1 \ldots \mathbf{c}_n)^T$ is a known matrix. Then

$$\frac{\partial \mathbf{h}(\mathbf{z}(t),t)}{\partial \mathfrak{z}} = \begin{pmatrix} \mathbf{c}_1^{\mathrm{T}} \frac{\partial \mathbf{z}}{\partial \mathfrak{z}_1} \\ \vdots \\ \mathbf{c}_n^{\mathrm{T}} \frac{\partial \mathbf{z}}{\partial \mathfrak{z}_n} \end{pmatrix}.$$

Differentiate once the jth row of (10.1) with respect to $\mathfrak{z}_k$ to obtain the differential equation

$$\dot{s}_{jk} = \frac{\partial f_j(\mathbf{z})}{\partial z_j} s_{jk}. \tag{10.7}$$

Since $\mathfrak{z}$ is a scaled version of $\mathbf{z}(t_0)$, the initial conditions for the sensitivity functions are

$$s_{jk}(t_0) = \frac{\partial z_j(t_0)}{\partial \mathfrak{z}_k} = \begin{cases} d_{jj} & \text{if } j = k, \\ 0 & \text{else.} \end{cases}$$

The sensitivity functions may then be obtained by considering a new *extended* state-space model consisting of (10.1) and of all differential equations (10.7) satisfied by the sensitivity functions. Müller's theorem turns out to be especially useful, as the extended state-space model is seldom cooperative, even if this is the case of the initial state-space model.

10.4.3 Using Taylor Models

In the case of Taylor models, the correction step is quite different from what has been described in Section 10.4.1. The measurement equation (10.2), (10.3) is rewritten as

$$\mathbf{h}(\mathbf{z}_{k+1}) + \boldsymbol{\delta} - \mathbf{y}(t_{k+1}) = 0. \tag{10.8}$$

Each Taylor model $\mathbf{T}_{\rho,k+1}^{(pr,l)}(\mathfrak{z})$ of $\mathscr{T}_{k+1}^{(pr)}$ is now considered separately and substituted for $\mathbf{z}_{k+1}$ in (10.8) to obtain

$$\mathbf{h}\left(\mathbf{T}_{\rho,k+1}^{(pr,l)}(\mathfrak{z})\right) + \boldsymbol{\delta} - \mathbf{y}(t_{k+1}) = 0. \tag{10.9}$$

For nonlinear systems, the left hand side of (10.9) is nonlinear even for a linear measurement equation, since the Taylor model $\mathbf{T}^{(pr,l)}_{\rho,k+1}(\mathfrak{z})$ is nonlinear. One knows that $\boldsymbol{\delta} \in [-\overline{\boldsymbol{\delta}}, \overline{\boldsymbol{\delta}}]$, thus, (10.9) may be solved for $\mathfrak{z}$ with an interval Newton method, which leads to a tightened domain interval $[\tilde{\mathfrak{z}}]$, hence $\mathfrak{z} \in [\tilde{\mathfrak{z}}]$. Here, the Krawczyk method is used. For $n > m$, (10.9) is under-determined and cannot be inverted. A simple approach is to solve (10.9) only for m variables (components of $\mathfrak{z}$), while considering the remaining $n-m$ variables as constant intervals.

This procedure is illustrated for a very simple linear example with $n = 2$ and $m = 1$. Consider the case when (10.9) is given by

$$\mathfrak{z}_1 + 0.5\mathfrak{z}_2 + \delta_1 - y_1(t_{k+1}) = 0, \tag{10.10}$$

with $[\mathfrak{z}_1] = [-1;1]$, $[\mathfrak{z}_2] = [-1;1]$, $y_1(t_{k+1}) = 0.9$, and $[\delta_1] = [-0.1;0.1]$. Now, (10.10) is first solved for $\mathfrak{z}_1$, the interval $[-1;1]$ being used for $\mathfrak{z}_2$. We have

$$[\tilde{\mathfrak{z}}_1] = y_1(t_{k+1}) - [\delta_1] - 0.5[\mathfrak{z}_2] = 0.9 - [-0.1;0.1] - 0.5[-1;1] = [0.3;1.5].$$

Now this result is intersected with the initial interval enclosure $[-1;1]$ resulting in

$$[\tilde{\mathfrak{z}}_1] = [0.3;1.5] \cap [-1;1] = [0.3;1].$$

Next, (10.10) is solved for $\mathfrak{z}_2$ with the new $[\mathfrak{z}_1]$

$$[\tilde{\mathfrak{z}}_2] = 2(y_1(t_{k+1}) - [\delta_1] - [\tilde{\mathfrak{z}}_1]) = 2(0.9 - [-0.1;0.1] - [0.3;1]) = [-0.4;1.4].$$

An intersection with the initial interval enclosure $[-1;1]$ results in

$$[\tilde{\mathfrak{z}}_2] = [-0.4;1.4] \cap [-1;1] = [-0.4;1].$$

Another possibility is to consider a sufficient number of previous measurements $\mathbf{y}(t \leq t_k)$ and the corresponding Taylor models of the right hand side of (10.9) to obtain the missing $n-m$ equations.

If no solution in $[\mathfrak{z}]$ can be found, then the corresponding Taylor model is inconsistent with the data and can be deleted.

As previously, the Taylor model is computed for a normalized domain box and written as

$$[\tilde{\mathfrak{z}}] \in \widetilde{\mathbf{T}}(\mathfrak{z}) = \tilde{\mathbf{c}} + \widetilde{\mathbf{D}}\mathfrak{z}$$
$$\text{with } \mathfrak{z}_i \in [-1;1], i = 1 \ldots n,$$

where $\tilde{\mathbf{c}}$ is the midpoint of $[\tilde{\mathfrak{z}}]$ and $\widetilde{\mathbf{D}}$ is a diagonal matrix with $\tilde{d}_{i,i} = \mathrm{rad}([\tilde{\mathfrak{z}}_i])$. The Taylor model $\mathbf{T}^{(c,l)}_{k+1}$ after the correction step is then given by

$$\mathbf{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z}) = \mathbf{T}^{(pr,l)}_{\rho,k+1}(\widetilde{\mathbf{T}}(\mathfrak{z})),$$

which defines the l-th Taylor model for the next prediction step:

$$\mathbf{T}^{(l)}_{\rho,k+1}(\mathfrak{z}) = \mathbf{T}^{(c,l)}_{\rho,k+1}(\mathfrak{z}).$$

The list of Taylor models at $k+1$ is then given by

$$\mathbf{z}_{k+1} \in \mathscr{T}_{\rho,k+1} = \left\{ \mathbf{T}^{(1)}_{\rho,k+1}(\mathfrak{z}), \mathbf{T}^{(2)}_{\rho,k+1}(\mathfrak{z}) \dots \mathbf{T}^{(L_{k+1})}_{\rho,k+1}(\mathfrak{z}) \right\}$$
$$\text{with} \quad \mathfrak{z}_i = [-1;1], \quad i = 1 \dots n \quad \text{with} \quad L_{k+1} \leq L^{pr}_{k+1} \leq L_{max} \ .$$

Interval Newton methods, like the Krawczyk method, involve the computation of the derivatives of $\mathbf{h}$ with respect to the initial states $\mathfrak{z}$ similar to (10.6) in the correction step for the method based on Müller's theorem. Since the Taylor model performs an expansion in the initial states $\mathfrak{z}$, the derivatives are easily obtained by calculating them for the Taylor model obtained from (10.9).

The correction step in combination with preconditioning is explained in [14], together with a correction step that involves consistency tests.

10.5 Simulation Results

For a comparison of the two approaches, consider the following nonlinear system

$$\begin{cases} \dot{x}_1 = & -p_3 x_1 - \dfrac{p_1 x_1}{1 + p_2 x_1} + p_4 x_2 \\ \dot{x}_2 = & p_3 x_1 - p_4 x_2 \end{cases}$$

with the initial value of the state $x_1(0) = 1$ and $x_2(0) = 0$.

Only x_2 is measured and the measurement equation at $t = t_k$ is

$$y_k = x_2(t_k) + \delta_k.$$

The parameters p_1, p_2 and p_3 are given by $p_1 = 1$, $p_2 = 1.2$ and $p_3 = 0.5$. The parameter p_4 is uncertain with $p_4 \in [0.1, 0.5]$. Thus, the extended state satisfies

$$\begin{cases} \dot{z}_1 = & -p_3 z_1 - \dfrac{p_1 z_1}{1 + p_2 z_1} + z_3 z_2 \\ \dot{z}_2 = & p_3 z_1 - z_3 z_2 \\ \dot{z}_3 = & 0 \end{cases} \tag{10.11}$$

with $z_1 = x_1$, $z_2 = x_2$, and $z_3 = p_4$. Measurements are assumed to take place every 2 s. The measurement uncertainty is given by $\delta(t_k) \in [-0.005, 0.005]$ for all t_k.

10.5.1 Results with Prediction Based on Müller's Theorem

10.5.1.1 Correction using Sivia

We used the Sivia algorithm combined with ImageSp to estimate the extended state vector. For this purpose, a coupled system of ODEs is built from (10.11)

$$\begin{cases} \dot{\underline{z}}_1 = -p_3\underline{z}_1 - \dfrac{p_1\underline{z}_1}{1+p_2\underline{z}_1} + \underline{z}_3\underline{z}_2 \\ \dot{\underline{z}}_2 = p_3\underline{z}_1 - \overline{z}_3\underline{z}_2 \\ \dot{\underline{z}}_3 = 0 \\ \dot{\overline{z}}_1 = -p_3\overline{z}_1 - \dfrac{p_1\overline{z}_1}{1+p_2\overline{z}_1} + \overline{z}_3\overline{z}_2 \\ \dot{\overline{z}}_2 = p_3\overline{z}_1 - \underline{z}_3\overline{z}_2 \\ \dot{\overline{z}}_3 = 0 \end{cases} \tag{10.12}$$

with $\underline{z}_1(0) = \overline{z}_1(0) = 1$, $\underline{z}_2(0) = \overline{z}_2(0) = 0$, and $\left[\underline{z}_3(0), \overline{z}_3(0)\right]$ dependent on the iteration in Sivia. Using (10.12), an inclusion function for $h(\mathbf{z}(t_k)) = z_2(t_k)$ is obtained.

The estimates for p_4 obtained at $t = 10\ s$ using the Sivia algorithm for various values of the precision parameter ε are provided in Table 10.1. As expected, reducing ε increases the computing time and the accuracy of the estimate. These results have been obtained using a single core of a two-processor, quad-core Intel Xeon CPU E5462 at 2.80 GHz with 6 144 KB cache and 64 GB RAM.

ε	time (s)	estimate for p_4
0.05	1.28	[0.125,0.35]
0.025	1.85	[0.1625,0.3125]
0.01	4.3	0.2[1718,8750]
0.005	8.28	0.2[3303,6875]
0.0025	15.77	0.2[4063,6251]
0.001	31.95	0.2[4546,5804]

Table 10.1: Estimates for p_4 obtained with Sivia at $t = 10\ s$ for various values of ε

10.5.1.2 Correction by Constraint Propagation

Since only the third initial condition is unknown, only the first-order sensitivity functions with respect to z_3 have to be evaluated. For that purpose, each ODE in (10.11) is derived with respect to z_3 to get

$$\begin{cases} \dot{s}_{13} = \left(-p_3 - \dfrac{p_1}{1+p_2 z_1} + \dfrac{p_1 p_2}{\left(1+p_2 z_1\right)^2} \right) s_{13} + s_{33} z_2 + z_3 s_{23} \\ \dot{s}_{23} = p_3 s_{13} - s_{33} z_2 - z_3 s_{23} \\ \dot{s}_{33} = 0 \end{cases} \tag{10.13}$$

with $s_{13}(0) = s_{23}(0) = 0$ and $s_{33}(0) = 1$. Müller's theorem may be used with (10.13) to compute an enclosure of the sensitivity function of the state with respect to the third initial condition, which is also p_4

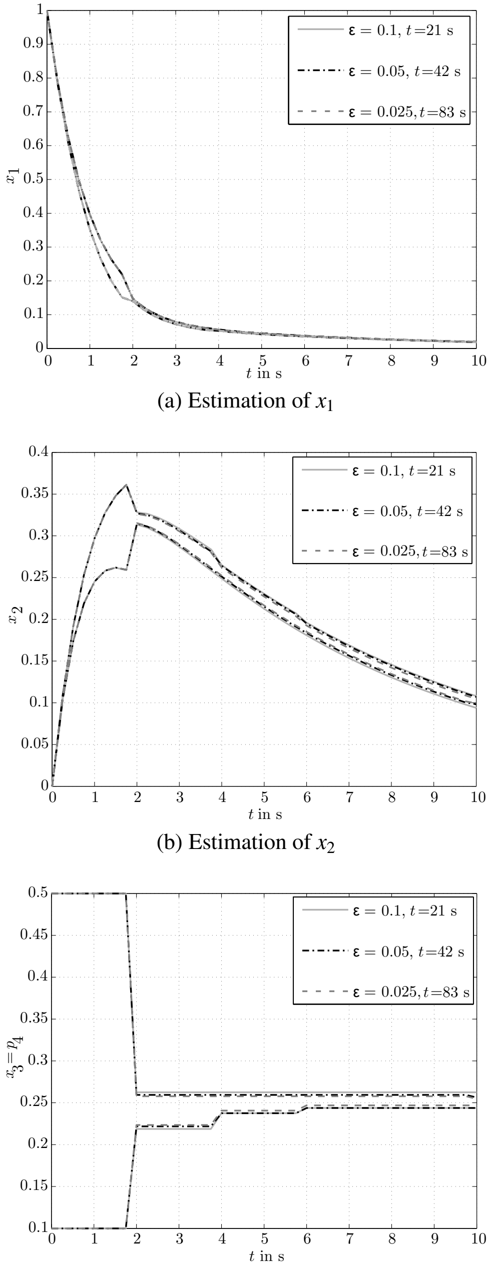
$$\begin{cases} \dot{\underline{s}}_{13} = \left(-p_3 - \dfrac{p_1}{1+p_2 \underline{z}_1} + \dfrac{p_1 p_2 \underline{z}_1}{\left(1+p_2 \overline{z}_1\right)^2} \right) \underline{s}_{13} + \underline{z}_2 \underline{s}_{33} + \overline{z}_3 \underline{s}_{23} \\ \dot{\underline{s}}_{23} = p_3 \underline{s}_{13} - \overline{s}_{33} \overline{z}_2 - \underline{z}_3 \underline{s}_{23} \\ \dot{\underline{s}}_{33} = 0 \\ \dot{\overline{s}}_{13} = \left(-p_3 - \dfrac{p_1}{1+p_2 \overline{z}_1} + \dfrac{p_1 p_2 \overline{z}_1}{\left(1+p_2 \underline{z}_1\right)^2} \right) \overline{s}_{13} + \overline{s}_{33} \overline{z}_2 + \underline{z}_3 \overline{s}_{23} \\ \dot{\overline{s}}_{23} = p_3 \overline{s}_{13} - \underline{s}_{33} \underline{z}_2 - \overline{z}_3 \overline{s}_{23} \\ \dot{\overline{s}}_{33} = 0 \end{cases}$$

This enclosure uses the fact that all quantities are positive, except $s_{23}(t) \leq 0$ for $t \geq 0$. The contractor described in Section 10.4.2 may then be employed in conjunction with `Sivia` to reduce the uncertainty on the initial value of the third component of the state.

Table 10.2 describes the evolution with time of the estimate for p_4 as a function of the precision parameter ε. Reducing ε again increases the accuracy at which the estimate is obtained, but the price to be paid is an increased complexity, since more ODEs are solved. Contrary to `Sivia`, a decent estimate is obtained even with the largest value of ε. The first and second measurements provide the most information about p_4, since for these measurements, the best decrease in the size of $[z_3]$ is observed. See also Figure 10.1. The same processor as in Section 10.5.1.1 has been used here.

Time	$\varepsilon = 0.2$	$\varepsilon = 0.1$	$\varepsilon = 0.05$	$\varepsilon = 0.01$
2 s	0.2[20488,57674]	0.2[24304,57668]	0.2[23906,57668]	0.2[24323,56828]
4 s	0.2[30813,57674]	0.2[34575,57668]	0.2[35411,57668]	0.2[40556,56828]
6 s	0.2[30813,57674]	0.2[36483,57668]	0.2[37597,57668]	0.2[44613,56828]
8 s	0.2[30813,57674]	0.2[36483,57668]	0.2[37597,57668]	0.2[44613,56828]
10 s	0.2[30813,57674]	0.2[36483,57668]	0.2[37597,57668]	0.2[44613,56250]
Comp.	1.0 s	2.13 s	3.47 s	17 s

Table 10.2: Evolution of the estimate for p_4 with `Sivia` for various values of the precision parameter ε as a function of the time and total computing time



(a) Estimation of x_1

(b) Estimation of x_2

(c) Estimation of p_4

Fig. 10.1: Comparison of the estimation results with `Sivia` for different values of the precision parameter ε used in the Müller method

10.5.2 Prediction and Correction Involving Taylor Models

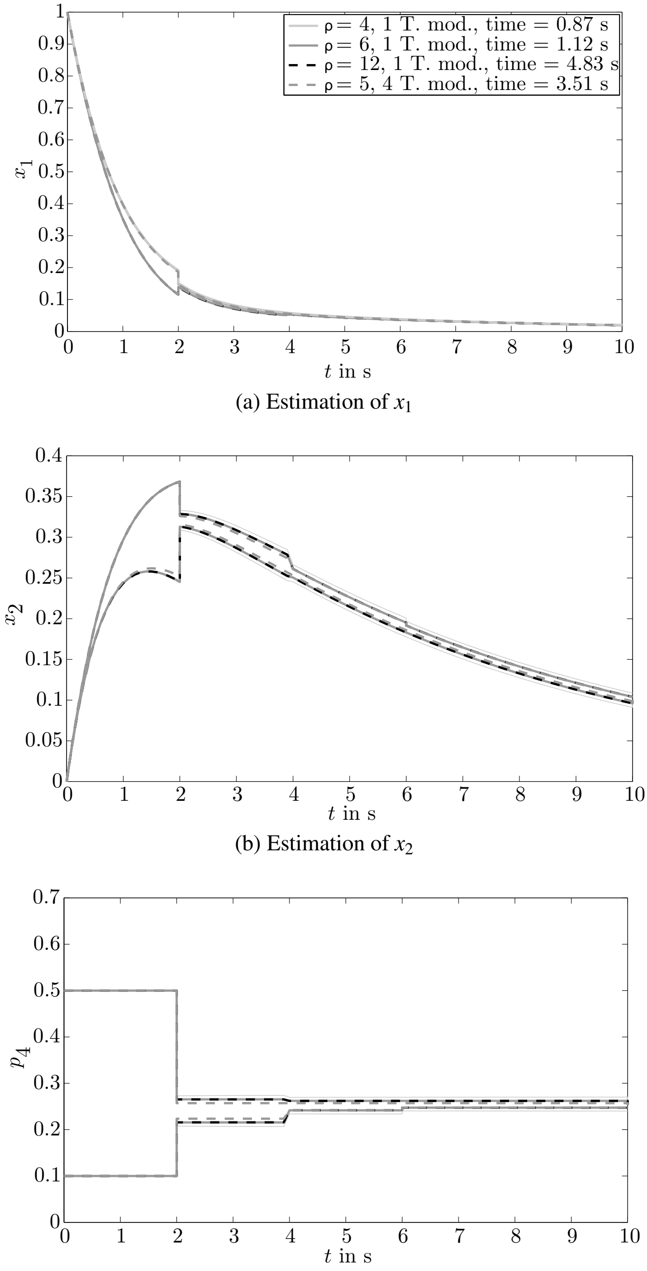
Estimation results for the approach based on Taylor models are shown for different orders and numbers L_{max} of Taylor models. In Figure 10.2 results for $\rho = 4$, $\rho = 6$, and $\rho = 12$ for a single Taylor model ($L_{max} = 1$, hence without splitting of the domain box) are depicted together with one result for $\rho = 5$ with 4 Taylor models ($L_{max} = 4$), hence with splitting of the domain box. In this section, all computations have been done on a Intel Centrino Core2 Duo T7300 at 2 GHz. In Table 10.3 the interval enclosures for the evolution of the estimate for p_4 as a function of time and total computing time are given. In Table 10.4 the interval enclosures for the estimated parameter p_4 at $t = 10\ s$ for various orders and numbers of Taylor models are presented. Increasing the order from 4 to 6 leads to an improvement with a slightly higher computation time. However if the order is increased to 12, the computation time increases drastically (from 1.12s to 4.83s) without tightening the enclosures in any significant manner. Further improvement of the estimation results is achieved only if splitting is applied as the results for $\rho = 5$ and $L_{max} = 4$ indicate. Even the computation time is lower than for $\rho = 12$ and $L_{max} = 1$.

Time t	$\rho = 4, L_{max} = 1$	$\rho = 6, L_{max} = 1$	$\rho = 12, L_{max} = 1$	$\rho = 5, L_{max} = 4$
2 s	0.2[07577 ,73053]	0.2[15534 ,65326]	0.2[15721 ,65149]	0.2[23622 ,57193]
4 s	0.2[33970 ,69719]	0.2[41587 ,62131]	0.2[41710 ,62005]	0.2[41941 ,57194]
6 s	0.2[33758 ,69931]	0.2[41585 ,62133]	0.2[41710 ,62005]	0.2[41941 ,57194]
8 s	0.2[40235 ,70064]	0.2[47399 ,62134]	0.2[47508 ,62005]	0.2[47786 ,57194]
10 s	0.2[40111 ,60466]	0.2[47399 ,53480]	0.2[47508 ,53377]	0.2[47786 ,53354]
Comp.	0.87 s	1.12 s	4.83 s	3.51 s

Table 10.3: Evolution of the estimate for p_4 for various values of ρ and L_{max} as a function of time, and total computing time

ρ	L_{max}	comp. time (s)	estimate for p_4
4	1	0.87	0.2[40111,60466]
6	1	1.12	0.2[47399,53480]
12	1	4.83	0.2[47508,53377]
5	4	3.51	0.2[47786,53354]

Table 10.4: Estimates obtained with Taylor models at $t = 10\ s$ for various values of ρ and numbers of Taylor models.



(a) Estimation of x_1

(b) Estimation of x_2

(c) Estimation of p_4

Fig. 10.2: Comparison of the estimation results for different orders and number of Taylor models.

10.6 Conclusions and Perspectives

Physics, chemistry (and most other experimental sciences) tend to produce continuous-time models whose outputs are nonlinear in their unknown parameters. When these models are in state-space form, they are almost always nonlinear in the extended state vector obtained by concatenating the state and parameter vectors. Most of the methods available for estimating this extended state vector are based on linearization or random exploration and cannot provide any guarantee as to their results. Since each of the parameters and state variables of such models has a concrete physical meaning, this is unfortunate. One would rather like to be able to characterize in some guaranteed way the set of all acceptable estimates given what is known of the uncertainty in the experimental data.

For quite some time, this has been completely out of reach, but a combination of advances in interval analysis, constraint propagation, and guaranteed integration of ODEs, together with a massive increase in computing power have now made this achievable, at least for small-scale models. This chapter has presented, in a coordinated manner, a variety of tools that can be used when bounds are available on the acceptable difference between the data and corresponding model output.

Guaranteed set estimation makes it possible to bypass the usual requirements of identifiability, observability, and persistency of excitation. These properties should nevertheless definitely contribute to the quality of the parameter and state estimates. More generally, the problem of experiment design for guaranteed parameter or state estimation is an interesting but still largely open question.

The main challenge is to increase the complexity of the models that can be studied with this type of guaranteed approach. To this end, it is necessary to use and possibly combine tools that struggle as efficiently as possible with the pessimism inherent to interval analysis and guaranteed integration and the curse of dimensionality. Contractors, which make it possible to eliminate parts of the search region without the need for bisections, and high-order Taylor models are among the most promising avenues for research.

References

1. Berz, M., Makino, K.: Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. Reliable Computing **4**, 361–369 (1998)
2. Chernousko, F.L.: Optimal guaranteed estimates of indeterminacies with the aid of ellipsoids. Engrg. Cybernetics **18**, 1–9 (1980)
3. Chernousko, F.L.: State Estimation for Dynamic Systems. CRC Press, Boca Raton, FL (1994)
4. Gennat, M., Tibken, B.: Simulation of uncertain systems with guaranteed bounds. In: 11th GAMM - IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics. Fukuoka, Japan (2004)
5. Gouzé, J.L., Rapaport, A., Hadj-Sadok, Z.M.: Interval observers for uncertain biological systems. Journal of Ecological Modelling (133), 45–56 (2000)

6. Hoefkens, J., Berz, M., Makino, K.: Efficient high-order methods for ODEs and DAEs. In: G. Corliss, C. Faure, A. Griewank (eds.) Automatic Differentiation : From Simulation to Optimization, pp. 341–351. Springer-Verlag, New-York, NY (2001)
7. Jaulin, L.: Nonlinear bounded-error state estimation of continuous-time systems. Automatica **38**, 1079–1082 (2002)
8. Jaulin, L., Kieffer, M., Didrit, O., Walter, E.: Applied Interval Analysis. Springer-Verlag, London (2001)
9. Jaulin, L., Walter, E.: Set inversion via interval analysis for nonlinear bounded-error estimation. Automatica **29**(4), 1053–1064 (1993)
10. Kieffer, M., Walter, E.: Guaranteed parameter estimation for cooperative systems. In: L. Benvenuti, A. De Santis, L. Farina (eds.) Positive Systems – LNCIS 294, pp. 103–110. Springer, Berlin (2003)
11. Kieffer, M., Walter, E.: Interval analysis for guaranteed nonlinear parameter and state estimation. Mathematical and Computer Modelling of Dynamic Systems **11**(2), 171–181 (2005)
12. Kieffer, M., Walter, E.: Guaranteed nonlinear state estimation for continuous–time dynamical models from discrete–time measurements. In: Preprints of the 5th IFAC Symposium on Robust Control Design (2006)
13. Kieffer, M., Walter, E.: Guaranteed estimation of the parameters of nonlinear continuous–time models: contributions of interval analysis. Int. J. Adap. Contr. Sig. Proc. (2010). Doi : 10.1002/acs.1194
14. Kletting, M.: Verified Methods for State and Parameter Estimators for Nonlinear Uncertain Systems with Applications in Engineering. Ph.D. thesis, Institute of Measurement, Control, and Microtechnology,University of Ulm, Germany (2009)
15. Kühn, W.: Rigorously computed orbits of dynamical systems without the wrapping effect. Computing **61**(1), 47–67 (1998)
16. Kurzhanski, A., Valyi, I.: Ellipsoidal Calculus for Estimation and Control. Birkhäuser, Boston, MA (1997)
17. Lin, Y., Stadtherr, M.A.: Validated solution of ODEs with parametric uncertainties. In: W. Marquardt, C. Pantelides (eds.) Proc. 16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering, *Computer Aided Chemical Engineering*, vol. 21, pp. 167 – 172. Elsevier (2006). DOI 10.1016/S1570-7946(06)80041-6. URL http://www.sciencedirect.com/science/article/B8G5G-4P37F2K-S/2/a22680956d2786784b23e88e9b272db6
18. Lohner, R.: Computation of guaranteed enclosures for the solutions of ordinary initial and boundary value-problem. In: J.R. Cash, I. Gladwell (eds.) Computational Ordinary Differential Equations, pp. 425–435. Clarendon Press, Oxford (1992)
19. Makino, K., Berz, M.: Suppression of the wrapping effect by Taylor model-based verified integrators: Long-term stabilization by preconditioning. International Journal of Differential Equations and Applications **10**(4), 353–384 (2005)
20. Makino, K., Berz, M.: Suppression of the wrapping effect by Taylor model-based verified integrators: Long-term stabilization by shrink wrapping. International Journal of Differential Equations and Applications **10**(4), 385–403 (2005)
21. Meslem, N., Ramdani, N., Candau, Y.: Guaranteed state bounding estimation for uncertain non linear continuous systems using hybrid automata. In: Proc. IFAC International Conference on Informatics in Control, Automation Robotics. Funchal, Madeira (2008)
22. Meslem, N., Ramdani, N., Candau, Y.: Interval observers for uncertain nonlinear systems. application to bioreactors. In: Proc. IFAC World Congress, pp. 9667–9672. Seoul, Korea (2008)
23. Moisan, M., Bernard, O., Gouzé, J.L.: Near optimal interval observers bundle for uncertain bioreactors. Automatica **45**(1), 291–295 (2009)
24. Müller, M.: Über das Fundamentaltheorem in der Theorie der gewöhnlichen Differentialgleichungen. Math. Z. **26**, 619–645 (1926)
25. Nedialkov, N.S., Jackson, K.R.: Methods for initial value problems for ordinary differential equations. In: U. Kulisch, R. Lohner, A. Facius (eds.) Perspectives on Enclosure Methods, pp. 219–264. Springer-Verlag, Vienna (2001)

26. Raissi, T., Ramdani, N., Candau, Y.: Set membership state and parameter estimation for systems described by nonlinear differential equations. Automatica **40**(10), 1771–1777 (2004)
27. Raissi, T., Videau, G., Zolghadri, A.: Interval observer design for consistency checks of nonlinear continuous-time systems. Automatica **46**(3), 518–527 (2010)
28. Rauh, A., Auer, E., Minisini, J., Hofer, E.P.: Extensions of ValEncIA-IVP for reduction of overestimation, for simulation of differential algebraic systems, and for dynamical optimization. Proc. Appl. Math. Mech. **7**(1), 1023,0011023,002 (2007). DOI 10.1002/pamm.200700022
29. Rauh, A., Hofer, E.P., Auer, E.: VALENCIA-IVP: A comparison with other initial value problem solvers. In: Proc. 12th GAMM - IMACS International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics (SCAN 2006), p. 36. IEEE Computer Society, Duisburg, Germany (2006). DOI 10.1109/SCAN.2006.47
30. Veres, S.M., Mayne, D.Q.: Geometric bounding toolbox. Tech. rep. URL http://www.sysbrain.com/gbt/
31. Walter, E., Kieffer, M.: Interval analysis for guaranteed nonlinear estimation. In: Proceedings of the 13th IFAC Symposium on System Identification (SYSID), pp. 259–270 (2003)
32. Walter, E., Kieffer, M.: Guaranteed nonlinear parameter estimation in knowledge–based models. Journal of Computational and Applied Mathematics **199**(2), 277–285 (2007)

Chapter 11
Quantifying Spacecraft Failure in an Uncertain Environment: the Case of Jupiter Europa Orbiter

Mehrdad Moshir

Abstract Study of the Outer Planets is considered as a high priority activity by the Planetary Science community. One candidate for the next Outer Planets Flagship Mission (OPFM —missions in the \$2B–\$4B range) is the *Jupiter Europa Orbiter* (JEO) concept. In this work, we address the interplay of various types of uncertainties to probe the possibility of characterizing the reliability of a proposed mission concept. By combining the *aleatory* characterization of spacecraft subsystems and the *epistemic* uncertainties of the Jovian environment we describe an approach for quantifying possible ranges of mission durations for a potential JEO concept. The work here illustrates the potential for probabilistic representations of epistemic uncertainties by introducing temporal correlations. In addition the effects of failure correlations among similar components in a spacecraft are incorporated to assess their impact on the failure likelihood.

11.1 Introduction

Over the past 40+ years spacecraft designers and engineers have learned the hard way that when proper attention is not paid to the uncertain nature of their operating environment rude awakening would be forthcoming with a vengeance. As a result of experience gained from successes as well as failures, a large body of experience has been accumulated [1–4]. Consequently space systems designers have devised rules for incorporating *safety margins* into design specifications to mitigate various system uncertainties [5]. As systems become more complex, the unregulated use of generous margins can eventually make a mission's cost non-competitive. As a consequence, other strategies for understanding the system robustness in a *quanti-*

Mehrdad Moshir
Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr., Pasadena, CA 91109, USA
e-mail: mehrdad.moshir@jpl.nasa.gov

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_11,

tative sense need to be considered. The longevity of missions such as Voyagers 1 & 2 demonstrates that when spacecrafts are not exposed to excessive radiation they can last a very long time by relying on standard design techniques. In contrast, intense radiation environments introduce a hazard that if not mitigated would lead to the early demise of the spacecraft, with the electronic components of the spacecraft being the prime causes of failure. It should be recalled that previous missions to the Jovian system have only made *sparse* measurements of the radiation levels – thus significant spatial and temporal *epistemic* uncertainties remain in modeling the radiation environment.

Among many interesting destinations within our solar system the Jovian neighborhood presents several worthy targets for visit. One such target is the moon Europa [6]. Previous spacecraft have produced a body of evidence that this moon may possess an ocean deep beneath its outer icy surface and there exist evidence of recent surface activity and changes, see Fig. 11.1. Previous missions to Jupiter have provided scientists and spacecraft engineers with a model of charged particle distribution in the Jovian system [7]. The important lesson learned over the years has been that for spacecraft the charged particle environment has very detrimental effects on the performance of various electronic components. An early failure of electronics components can lead to the premature termination of a mission if proper redundancy and shielding measures have not been incorporated in the design. Since spacecraft live a fairly autonomous life during most of their operation the heart and brain of the vehicle must be well shielded and their lifetime well understood.

In this work frequent reference to Total Ionizing Dose (TID) will be made. As a charged particle traverses matter it ionizes the material and loses energy as it

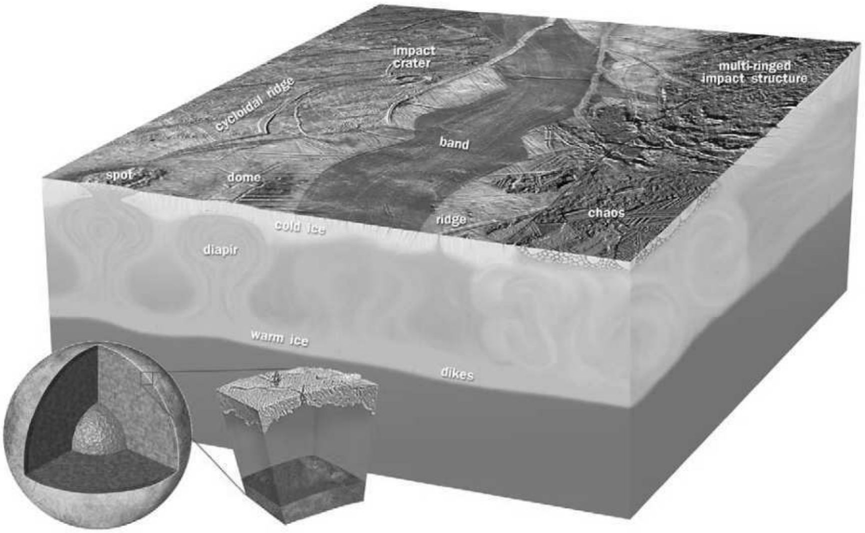


Fig. 11.1: Potential icy surface and ocean structure on Europa

progresses, the quantification of this effect is via the Linear Energy Transfer (LET) function defined to be $\frac{dE}{dx}$ –which is clearly material dependent. The number of charged particles in a given energy range dE crossing a unit area in unit time is denoted by particle flux $\phi(E,t)dE$. Over a finite time interval the integral of flux over time is denoted as the fluence, $\Phi(E)dE$. Combined with the LET the Ionizing Dose can be computed from $\int_E \Phi(E)LET(E)dE$. The unit of measurement is a Rad, defined to be equivalent to deposition of 100 ergs of energy into one gram of matter. In electronics design process the parts have "spec sheets" provided by the vendor indicating capabilities such as hardness with respect to radiation, typically expressed in KRads and occasionally in MRads .

References will be made to Radiation Design Factor (RDF), the following sets the context for its use. The RDF arose from the conventional approach to design, for example [8]; consider a part that is advertised to have a radiation hardness of say h_0, and suppose that part will operate in an environment that can deliver a total ionizing dose of D_0 (in reference to standard 100 mils aluminum). The spacecraft designers implemented a radiation shielding architecture around the component so that *no more than* h_0/S ionizing radiation would reach the component. The factor S is referred to as the RDF. The purpose of the factor S was multi-fold; the level D_0 corresponding to the predicted environment came from a model that could have major uncertainties; in this context S can be considered similar to a model uncertainty factor (MUF). Furthermore the calculations for shielding were not exact, and in case the designed shielding was not sufficient the factor S could potentially remedy the shielding uncertainty. There was also the possibility that the radiation hardness of the component might not be what it was believed to be; again the factor S could mitigate that concern. Typically an RDF of 2-3 has been used in many circumstances. Obviously the higher the S factor the more "resilient" the system gets, but shielding comes with a significant penalty in mass and since higher lift capabilities of launch vehicles come with a steep price premium there needs to be a more balanced and quantitative approach to design.

Previous missions have been able to use standard design techniques with an RDF of 2-3 because the expected radiation environments have not been excessive and there has been no need to study the lifetime problem from a probabilistic point of view. In the case of Europa the radiation environment is many times harsher than any mission has experienced; use of standard design techniques will result in such a large shielding mass that precludes carrying any science payload! Thus the need arises to study the lifetime problem probabilistically.

From a probabilistic point of view managing risks associated with design decisions in the previously described context is a very challenging undertaking due to the abundance of uncertain or insufficient information. Sometimes the lack of concrete evidence leads us to adopt expert opinion, use of expert opinion in a probabilistic sense is a topic worthy of monographs and is not directly addressed here but observations will be made in the course of this work. Common sense also informs us to consider the potential of correlations amongst the system elements. Depending on the context, correlations can lead to either worse or better results than the uncorrelated case. Computationally the incorporation of correlations into daily engineering considerations is fraught with difficulty due to several reasons. The curriculum for most practitioners does not include a detailed study of correlated random processes, thus the lack of familiarity with the methods precludes their consideration. Furthermore, tools to easily allow the use of multidimensional correlated random processes are not readily accessible. The new work described here incorporates correlation among elements, whether physical components or natural phenomena, and will investigate the impact of such assumptions on the final result.

The reliability and longevity of a flight system is affected by many effects such as natural aging, temperature variations, operational usage and duty cycles, as well as the effects of radiation environment. In this work the probabilistic estimate for the lifetime due to the radiation environment alone is assessed, specifically due to TID-induced failures. Other non-TID radiation-induced effects as well as natural aging, temperature induced degradation, mode of operation and duty cycles also do have a bearing on the lifetime estimate, but not included in this analysis. The historical data on the lifetime of spacecraft that have not been exposed to significant TID indicates that the engineering de-rating approaches and attendant design processes work well for benign radiation environments (for example Voyagers 1 & 2). For a spacecraft that departs Earth for Jupiter, it can be assumed to be robust to failure due to standard design techniques during its multi-year cruise phase. It is in the Jovian radiation environment that the design approaches need to take into account the shortened lifetime of the components and implement a strategy that allows the spacecraft perform a long duration science tour.

One high level abstraction and summary of the flight system is commonly the Master Equipment List (MEL) [9]. This abstraction includes enumeration of flight system elements, their masses, their multiplicities and their expected radiation hardnesses. The model described here utilizes the estimates from the MEL for the electronic circuit TID capabilities and their shielding levels (with an RDF of 2). The model generates probabilistic scenarios for the *whole flight system* TID capability. This capability is then compared with a stochastic characterization of the radiation exposure level at Europa to determine a prediction of the lifetime probability for a set of model parameters. The reference radiation design point is defined by the TID (for a 100 mils Al shell) at Europa Orbit Insertion (EOI) and at EOI plus 105 days; 1.65Mrad and 2.9Mrad, respectively.

The approach for modeling described in this work has been geared towards future extensibility of using test-based radiation hardness probability distributions. Such distributions may be presented in tabular form or as parameterized fits to functions that may not be amenable to analytical study. With these future considerations in mind the model has been implemented as a numerical method for Latin Hypercube sampling of the probability phase space. One important point to note is that in the engineering domain there is a predominance of easy to use tools such as *Microsoft Excel* for initial investigation of problems. These types of computations eventually may be best performed using more sophisticated computational capabilities such as those provided by *Matlab*, *C*, *Fortran*, *etc*. The methods discussed here were implemented as an *Excel* program that requires a commercial plug-in [10] for performing the Monte Carlo calculations.

The question of estimating the lifetime of a proposed mission to Europa has been considered recently by a method that relies on the approach of modeling parts *counts* and hardnesses derived from test-based expert judgment for two scenarios for the environment, uncorrelated or fully correlated radiation doses [11].

The work described here goes beyond the characterization that was described in [11]. This work employs a description of the flight system based on the MEL and also introduces a new approach to characterize the inherent *epistemic* uncertainty in the environment at Europa, the latter approach allows assessment of a continuum of radiation environment temporal correlations. The methodology allows assessment of more scenarios and assumptions for the effects of the radiation environment on the flight system.

The discussion of system characterization in the following sections is an example of *model-based systems engineering*, this approach to systems engineering is recent and is beginning to be recognized by many complex space missions as a necessary step to enhance mission success. For example, the Space Interferometry Mission (SIM Lite) concept required to demonstrate the capability to detect *Earth-mass* planets in the *habitable zones* of *nearby* solar systems. The systems engineering topics on this mission concept were similar to other projects; however the precision requirements, which are beyond anything ever attempted, required judicious use of validated models to accomplish the objectives. To support this objective a combined model of the spacecraft, the instrument, the behavior of distant planetary systems and typical operating scenarios was developed and performance results for five-year mission concepts under many scenarios were derived to demonstrate the capability of SIM Lite concept to detect *Earth-mass* planets. In many ways *model-based systems engineering* found a true application in the SIM Lite concept [12].

11.2 Model Description

To place reasonable bounds on the likelihood of survival in an uncertain environment the problem needs to be broken down into manageable elements, with each element treated probabilistically and finally reassembling the elements into the full picture. Model construction has two facets, one uses empirically determined probability distributions for electronic component hardnesses, while the other invokes a radiation model (using sparsely sampled data) for the Jovian environment and enhances it by adding an element of uncertainty by hypothesizing the potential for temporal correlations in the expected radiation levels. The first facet is therefore expressible by *aleatory* uncertainties, whereas the second is an expression of *epistemic* uncertainties.

The radiation dominated reliability concept invokes several notions. First notion, (a), is that for the full flight system to operate gainfully all of its subsystems must be operating; once *any* subsystem (as well as its redundant counterpart) has failed then the full system has failed. The other notion, (b), is that for each subsystem (and its redundant counterpart) to operate, *all* of its radiation sensitive elements must be in operating condition; when any one of the radiation sensitive elements of a subsystem has failed, the subsystem has failed. One last notion, (c), is that each electronic circuit is assembled from a small number of notional "functional components", and that each notional component will fail to operate once it has received some TID threshold. It is not known *a priori* with certainty what that TID threshold is for a specific functional component. Empirical evidence indicates that a sample of identical components possesses a range of such hardness thresholds that can be represented by a probability distribution that many practitioners approximate by a function similar to a lognormal probability distribution, $f_{LN}(\mu, \sigma)$. This adopted distribution is only to facilitate the computations; as noted in previously the actual distributions will more than likely not have simple forms such as a lognormal. When a system is decomposed into many notional components it becomes possible to define a probability distribution for the system TID capability –this characterization is *aleatory* and derived from empirical evidence.

The other component of the model is the characterization of the radiation environment. The analysis of data from previous missions has led to a model for the mean value of radiation environment with a spread that depends on the Jovian distance R_J [7]. Due to uncertainties in the physics of models and lack of sufficient data it may become tempting to treat the environment as a random variable. However, one must be cognizant of possible temporal correlations in the environment. There are indications in the radiation data from the Galileo mission [13] that temporal correlations in the observed electron fluxes exist. Ideally a power spectral density of the expected radiation fluctuations could shed significant light on the problem but currently no power spectral density for the radiation environment exists. This study adopts the approach of generating temporally correlated radiation dosage histories using the published mean and variances of electron fluxes [7], with details that will be described later.

The time at which the randomly generated radiation dosage exceeds the TID capability of the system gives the lifetime. The "random" nature of the environment and the expected TID capability of the system lead to a probability distribution for the lifetime; from this probability distribution the likelihood of survival at EOI+105 days or any other number of days beyond EOI can be derived.

The remainder of the discussion is structured as follows: a description of the usage of lognormal distribution is given, followed by characterization of spacecraft radiation hardness capability –based on empirical evidence and best expressed by *aleatory* uncertainties. Next the characterization of the random TID histories which are affected by *epistemic* uncertainties is discussed. Finally a comparison of spacecraft capability and dosage history is given, from which the lifetime probability distribution is derived.

Usage of Lognormal Distribution in the Model

For the circuit hardness discussion that follows, as well as the radiation environment characterization, the lognormal distribution will be invoked frequently. To be clear and to remove any confusion about the lognormal distribution parameter usage in this work we illustrate with an example. Imagine that a sample of N components from a given lot is exposed to radiation and the TID at failure for each one is recorded as $\{d_1, d_2, \ldots, d_N\}$. We define the mean hardness of the sample by $\mu = \frac{1}{N}\sum_{i=1}^{N} d_i$ and its variance by $\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(d_i - \mu)^2$.

Now suppose it is observed that natural logarithm of the dosages at failure $\{ln(d_1), ln(d_2), \ldots, ln(d_N)\}$ follows a Normal distribution. Then the probability distribution associated with the component hardness can be written as

$$f_{LN}(z, \mu', \sigma') = \frac{1}{z\sqrt{2\pi\sigma'}} e^{-\frac{1}{2}(\frac{\ln z' - \mu'}{\sigma'})^2} \tag{11.1}$$

Where z is a possible realization of the component hardness, and new parameters μ' and σ'^2 are defined by

$$\mu' = \ln\mu - \frac{1}{2}\ln(1 + \frac{\sigma^2}{\mu^2}) \tag{11.2}$$

$$\sigma'^2 = \ln(1 + \frac{\sigma^2}{\mu^2}) \tag{11.3}$$

For example if such a hypothetical sample has a mean value (μ) of 1Mrads and a standard deviation (σ) of 0.2Mrads, then the probability distribution for the sample can be written as $f_{LN}(z, -0.0196, 0.198)$. Suppose a random sample is taken from this distribution and expressed in Mrads, such a draw could be considered a random representative from the lot under consideration.

The important observation is that only two parameters fully characterize the distribution, μ and $\frac{\sigma}{\mu}$. The opinion of electronic device experts [14] points to the ratio $\frac{\sigma}{\mu}$ as being a significant discriminator among electronic parts. It is defined as the coefficient of variation, C_{OV}. In this work we extend its usage to devices consisting of parts and use the same terminology here while bearing in mind that in this case it refers to a functional component and not a part.

One aspect of "spec sheets" for electronic parts is their conservative nature of the estimates; experience indicates that if a part is advertised as having a radiation hardness of μ Mrads, to compensate for the typical conservativism, the estimate should be multiplied by a prudent scale factor of SF.

Using scale factor SF, coefficient of variation C_{OV} and a nominal rating μ, the parameters μ and σ change to $\mu \rightarrow SF\ \mu$ and $\sigma \rightarrow C_{OV}\ SF\ \mu$. These new values are what are used to calculate μ' and σ' for the *lognormal* distribution.

For a critical application such as an OPFM a program of radiation testing for vulnerable parts is planned to reduce uncertainties and biases that may result from expert judgement alone. After such tests are performed the scale factor *SF* would eventually be set to unity since the mean value from radiation testing is the true representative mean rating of components.

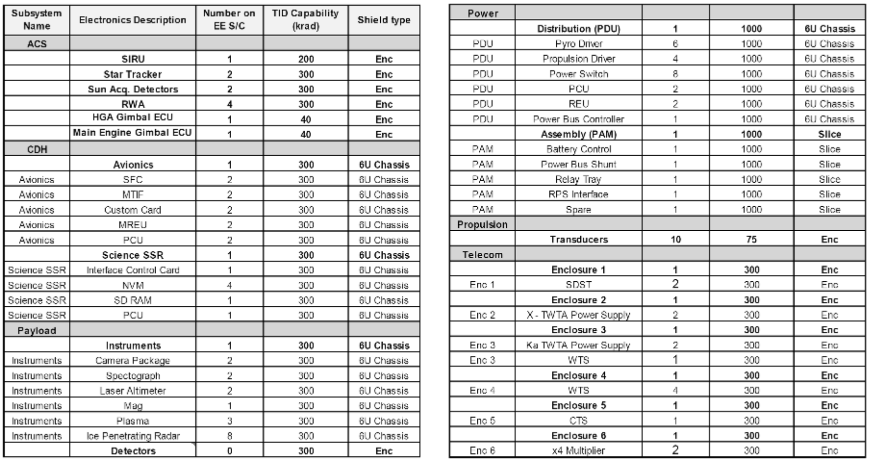
It is worthwhile to note that one other weapon in the hands of system designers is the use of "spot shielding" of vulnerable parts in a circuit. The possibility of using spot shielding for weak components can be translated into an additional *SF* in the model. Another method to improve on system capability is to "cherry pick" the components from a given lot; when this is performed, it can lead to a reduction of the C_{OV} as all parts will have very close capabilities.

As an engineering trade tool turning both *SF* and C_{OV} into variables in the model allows performing analyses to assess the effects of spot shielding and the required C_{OV} to reach specific reliability goals. In fact since both *SF* and C_{OV} currently represent expert judgement, they should properly be considered as elements with *epistemic* uncertainty and represented by probability distributions that reflect the state of belief of the user in the judgement of that expert. In the numerical framework we have discussed, such a characterization of the expert judgement via probability distributions is easily feasible but not undertaken for the sake of simplicity.

11.3 Characterization of Flight System by *Aleatory* Uncertainties

In Section 11.1 reference was made to the MEL that describes the counts of various subsystems of the spacecraft. An abbreviated typical MEL may look similar to that shown in Fig. 11.2. To illustrate the approach this MEL will be used to create the flight system TID capability with a given shielding strategy.

As can be seen in Fig. 11.2 each subsystem consists of several notional circuit boards. For modeling the notional electronic boards, each one from the MEL is decomposed into a different number of "functional components" from a small set that consists of digital logic/clocking, LVDS I/F, DC/DC converter, voltage regulator, memory, imaging array, optical encoder, PZT sensor, and other individual elements such as electromechanical and "radiation hard by design" components. Information from domain expertise [14] about reasonable values to assume for such components leads to adopting a notional set of parameters that are shown in Table 11.1. These values have been used in the ensuing computations. We also note that the scale factor *SF* and C_{OV} are parameters that could be varied to probe various scenarios because spot shielding and careful selection of components become equivalent to changing the scale factor *SF* and C_{OV}.



Subsystem Name	Electronics Description	Number on EE S/C	TID Capability (krad)	Shield type
ACS				
	SIRU	**1**	**200**	**Enc**
	Star Tracker	**2**	**300**	**Enc**
	Sun Acq. Detectors	**2**	**300**	**Enc**
	RWA	**4**	**300**	**Enc**
	HGA Gimbal ECU	**1**	**40**	**Enc**
	Main Engine Gimbal ECU	**1**	**40**	**Enc**
CDH				
	Avionics	**1**	**300**	**6U Chassis**
Avionics	SFC	2	300	6U Chassis
Avionics	MTIF	2	300	6U Chassis
Avionics	Custom Card	2	300	6U Chassis
Avionics	MREU	2	300	6U Chassis
Avionics	PCU	2	300	6U Chassis
	Science SSR	**1**	**300**	**6U Chassis**
Science SSR	Interface Control Card	1	300	6U Chassis
Science SSR	NVM	4	300	6U Chassis
Science SSR	SD RAM	1	300	6U Chassis
Science SSR	PCU	1	300	6U Chassis
Payload				
	Instruments	**1**	**300**	**6U Chassis**
Instruments	Camera Package	2	300	6U Chassis
Instruments	Spectograph	2	300	6U Chassis
Instruments	Laser Altimeter	2	300	6U Chassis
Instruments	Mag	1	300	6U Chassis
Instruments	Plasma	3	300	6U Chassis
Instruments	Ice Penetrating Radar	8	300	6U Chassis
	Detectors	**0**	**300**	**Enc**
Power				
	Distribution (PDU)	**1**	**1000**	**6U Chassis**
PDU	Pyro Driver	6	1000	6U Chassis
PDU	Propulsion Driver	4	1000	6U Chassis
PDU	Power Switch	8	1000	6U Chassis
PDU	PCU	2	1000	6U Chassis
PDU	REU	2	1000	6U Chassis
PDU	Power Bus Controller	1	1000	6U Chassis
	Assembly (PAM)	**1**	**1000**	**Slice**
PAM	Battery Control	1	1000	Slice
PAM	Power Bus Shunt	1	1000	Slice
PAM	Relay Tray	1	1000	Slice
PAM	RPS Interface	1	1000	Slice
PAM	Spare	1	1000	Slice
Propulsion				
	Transducers	**10**	**75**	**Enc**
Telecom				
	Enclosure 1	**1**	**300**	**Enc**
Enc 1	SDST	2	300	Enc
	Enclosure 2	**1**	**300**	**Enc**
Enc 2	X - TWTA Power Supply	2	300	Enc
	Enclosure 3	**1**	**300**	**Enc**
Enc 3	Ka TWTA Power Supply	2	300	Enc
Enc 3	WTS	1	300	Enc
	Enclosure 4	**1**	**300**	**Enc**
Enc 4	WTS	4	300	Enc
	Enclosure 5	**1**	**300**	**Enc**
Enc 5	CTS	1	300	Enc
	Enclosure 6	**1**	**300**	**Enc**
Enc 6	x4 Multiplier	2	300	Enc

Fig. 11.2: List of subsystems, their circuits and TID capabilities from the MEL for a notional JEO flight system

Type	**ID#**	**Nominal rating** (μ)	**Scale factor** (SF)	C$_{OV}$
Digital	1	1	1.5	.1 - .4
LVDS I/F	2	.5	1.5	.1 - .4
DC/DC	3	.3	1.5	.1 - .2
Volt Reg	4	.3	1.5	.1 - .2
Memory	5	.5	1.5	.1 - .2
Imaging array	6	1	1.5	.1 - .4
Optical encoder	7	.04	1.5	.1 - .2
PZT sensor	8	.075	1.5	.1 - .2
Electromechanical	9	.5	1.5	.1 - .4
RadHard by design	10	1	1.5	.1 - .4

Table 11.1: Parameters used for modeling various functional components chosen in the circuit hardness computations

11.3.1 Methodology Description for a Notional Board

For illustration of the computational method, an example of one notional board will be discussed in detail. Other boards follow the same approach but with different component counts and parameters. The example board has a very simple-minded abstraction of the functions of the SRU (stellar reference unit –aka star tracker), a cartoon of it is seen in Fig. 11.3. The decomposition shown here is not intended to be a one to one identification of what an eventual SRU would be like; rather it is intended to capture the typical functions that such a SRU may have.

To model the radiation hardness of this board six different log normal distributions are needed. To simplify the notation, the *lognormal* distribution corresponding to, for example the digital/clocking component, is written as $f_{LN}(\xi, \#1)$ where "#1"

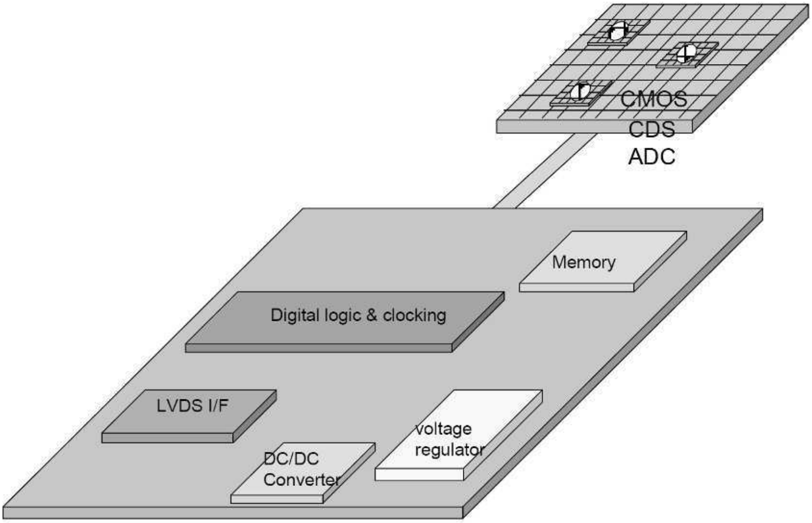


Fig. 11.3: Cartoon of a notional board using some of the functional components from Table 1

is a reference to the parameter in column 2 (ID#) from Table 11.1, and likewise for other components. So for the notional board shown in Fig. 11.3 we represent it mathematically as the random set

$$\{f_{LN}(.,\#1), f_{LN}(.,\#2), f_{LN}(.,\#3), f_{LN}(.,\#4), f_{LN}(.,\#5), f_{LN}(.,\#6)\} \tag{11.4}$$

For a board described as above an *aleatory* radiation hardness realization is denoted by the list $l = \{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6\}$ where each of ξ_j denotes a random draw from its corresponding distribution $f_{LN}(.,\#j)$.

According to Section 11.2, the board fails if any one of its functional components fails. The dosage at which this occurs is given by the *smallest* value in the list l. For this board the radiation hardness, h, is therefore.

$$h = \min(\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6) \tag{11.5}$$

For simple probability distributions and uncorrelated variables the probability distribution for h, $P(. > h)$, can be derived using the techniques of order statistics [15]. However in practice the distributions are derived empirically and numerical methods such as Monte Carlo need to be employed to compute $P(. > h)$, as will be done in the remainder of this work.

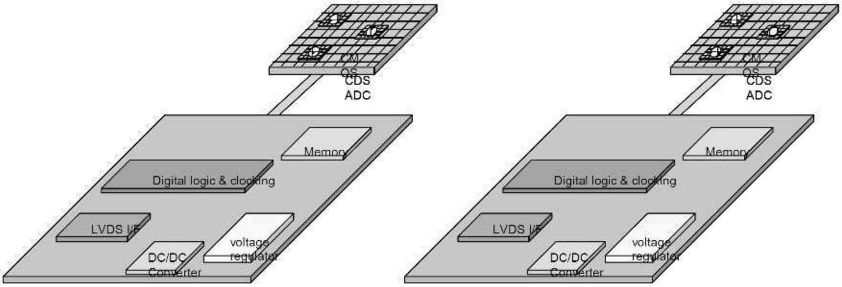


Fig. 11.4: Two identical boards with identical components that may have hardness correlations between like components

11.3.2 Methodology for Redundancy and Component Correlations

To illustrate how the model treats circuit redundancy we consider the cartoon shown in Fig. 11.4 which shows two identical notional boards where they use identical components. The same random number generation mechanism can be used as before to represent the circuits by two sets of the forms

$$\{f_{LN}(.,\#1), f_{LN}(.,\#2), f_{LN}(.,\#3), f_{LN}(.,\#4), f_{LN}(.,\#5), f_{LN}(.,\#6)\}_1 \tag{11.6}$$

$$\{f_{LN}(.,\#1), f_{LN}(.,\#2), f_{LN}(.,\#3), f_{LN}(.,\#4), f_{LN}(.,\#5), f_{LN}(.,\#6)\}_2 \tag{11.7}$$

one for each circuit. A numerical approach easily allows accounting for possible correlations of *component* hardnesses –such a scenario can come about because of “cherry picking” or the components having come from the same manufacturing lot. Of course when the same type of component is used in other circuits the hardness correlation of a given component can extend beyond identical boards. To simplify the calculations here the correlation is confined to the same circuit type. Extending the model to incorporate this extra correlation to all circuits that use the component is not a complex operation.

For each board (11.5) applies and the configuration radiation hardness is representable by the random set $\{h_1, h_2\}$. To illustrate correlation in component hardnesses suppose the components of the type with ID #1 that are used in the circuits have correlated hardnesses. Random number generation leads to a pair of hardnesses for the two components, one for each circuit, $\left\{\xi_1 \in f_{LN}(.,\#1), \xi_1' \in f_{LN}(.,\#1)\right\}$, with the condition that

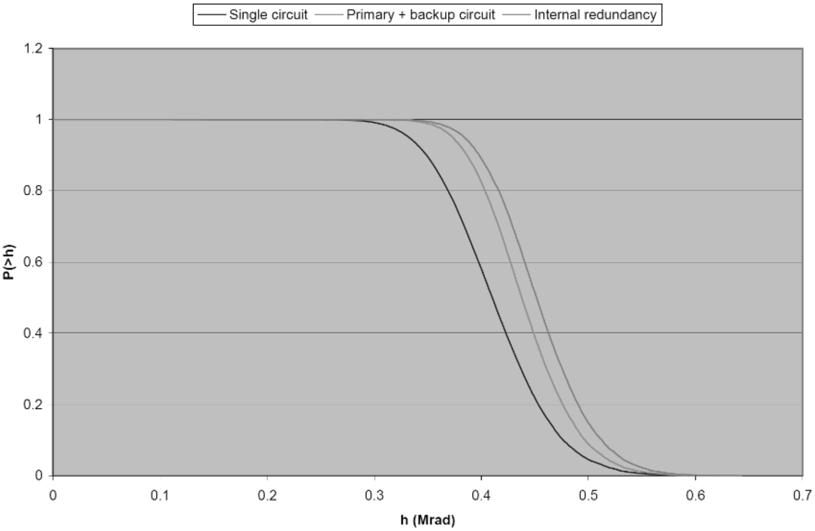


Fig. 11.5: The radiation hardness distribution for a single board, two boards (representing a primary and a backup). Also shown is a single board that has *internal* redundancy

$$\rho_{12} = \frac{cov(\xi_1, \xi_1')}{\sqrt{Var(\xi_1)}\sqrt{Var(\xi_1')}} \tag{11.8}$$

is a non-zero value. It is straightforward to compute the hardnesses for each circuit as before, with

$$h = \min(\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6)\ ,\ h' = \min(\xi_1', \xi_2', \xi_3', \xi_4', \xi_5', \xi_6') \tag{11.9}$$

With this construction the two boards develop correlated hardnesses because now ξ_1 and ξ_1' are correlated. The intuitive expectation is that two circuits should provide more hardness than one. The parameter that describes the benefit of two circuits is defined by the larger of the two board hardnesses

$$H = \max(h, h') \tag{11.10}$$

A plot of the distribution functions for H defined by (11.10) and h defined by (11.5), should show that H extends to the right of h. This is illustrated in Fig. 11.5. Figure 11.5 also includes a curve that indicates the hardness of *one* board but with *internal* redundancy within the same board for the *weakest* components of that board. The curve shows that internal redundancy within a board provides more hardness than two boards each without internal redundancy. However, in general designing boards with complex internal redundancy will be more costly to the system than using two simpler designed boards.

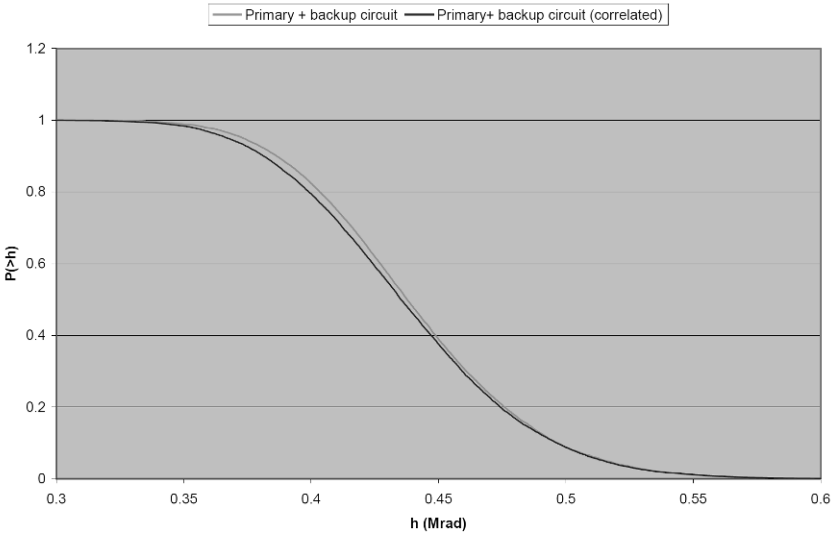


Fig. 11.6: Same configuration as in Fig. 5 but with correlation of component hardnesses included for two boards (primary + backup)

The illustration shown in Fig. 11.5 used components that were uncorrelated in hardness. The consequence of component hardness correlation for a primary and its backup circuit is shown in Fig. 11.6 for the same example board. In this case the components of same kind on the primary and backup were set to have a correlation coefficient of 25% according to (11.8). Figure 11.6 shows hardness for the primary + backup boards is reduced when there is a correlation between like components. For the example board shown the correlation effect is not extreme for a ρ_{12} as high as 25%.

11.3.3 Radiation Shielding and Hardness Capability

The discussion in Section 11.1 referred to the use of radiation shielding as an approach to protect the electronic systems and to prolong their lives. Historically at JPL a set of Design Principles [5] have been developed that capture the results of lessons learned from various missions. One parameter from the Design Principles is the Radiation Design Factor (RDF). Its intent is to accommodate the *epistemic* uncertainties related to the environment. In addition the design process for electronic circuits relies on a set of Worst Case Analysis (WCA) methods that provide a performance margin to the developer of the circuit elements, as the adjective “worst” indicates. Due to differing radiation hardnesses of different components a *normal-*

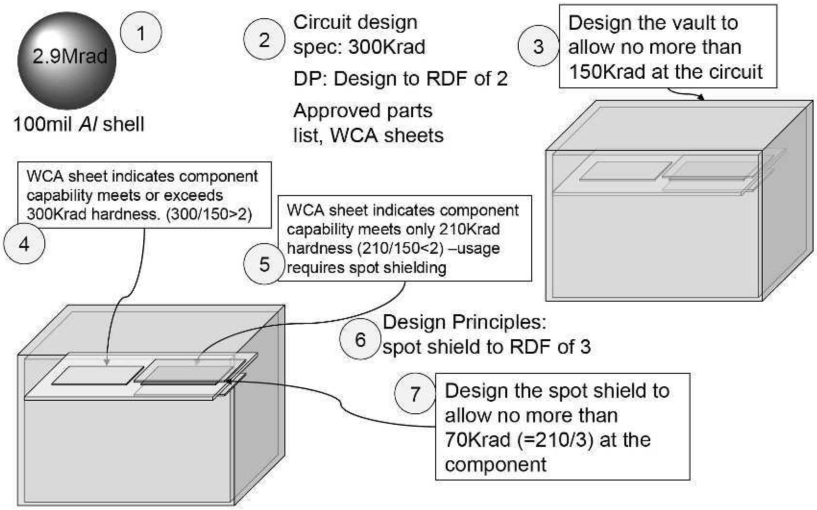

1
2.9Mrad
100mil Al shell
2
Circuit design spec: 300Krad
DP: Design to RDF of 2
Approved parts list, WCA sheets
3
Design the vault to allow no more than 150Krad at the circuit
4
WCA sheet indicates component capability meets or exceeds 300Krad hardness. (300/150>2)
5
WCA sheet indicates component capability meets only 210Krad hardness (210/150<2) –usage requires spot shielding
6
Design Principles: spot shield to RDF of 3
7
Design the spot shield to allow no more than 70Krad (=210/3) at the component

Fig. 11.7: The method of scaling to express circuit hardness in terms of equivalent external environment dosage

ization to the radiation field *external* to the shielding is required. The illustration of the process of using the RDF and WCA methods is shown in Fig. 11.7.

The circuit hardnesses that are evaluated according to the recipes in Sections 11.3.1 and 11.3.2, refer to the TID level the board receives. Boards are shielded from the radiation environment by being enclosed in shield types shown in the MEL. The Design Principles [5] require that an RDF of 2 is applied when designing the shielding, where

$$RDF = \frac{\text{Design Capability}}{\text{Expected local radiation received @design point}} \tag{11.11}$$

for each circuit the TID design capability is defined by the MEL. A value of RDF=2 therefore yields the level of radiation that is allowed at the circuit local environment at the reference design point. In the case of the proposed JEO mission, the design point is taken to be 2.9Mrads @EOI+105 days (100mils Al shell). The shielding process ensures that the proper type of shielding is designed so that each circuit receives no more radiation than

$$\frac{\text{Design capability}}{RDF}. \tag{11.12}$$

Therefore each unit of TID at the circuit corresponds to a *larger* level of TID *external* to the shield (expressed in 100mils Al shell). The scale factor is α given by

$$\alpha = RDF \frac{\text{TID @Design point}}{\text{Design capability}}. \tag{11.13}$$

Probabilistic radiation hardness curves described in Sections 11.3.1, 11.3.2 need to be properly scaled by the appropriate α factor that corresponds to the board's design capability. We denoted the probability distribution for a circuit's hardness as $P(. > h)$ in Sections 11.3.1, 11.3.2 (such as functions seen in Figs. 11.5, 11.6.) Board hardness is now expressed in terms of dosage by a rescaling of the argument of the probability distribution P, namely $P(. > \alpha\ h)$. This individual rescaling is applied to all circuits separately so they can all be treated on equal footing of being in a similar environment. This process of scaling, including additional scaling needed due to spot shielding was illustrated in Fig. 11.7. To allow for generality, in the model every circuit in the MEL has its own α factor, even though almost all circuits within a subsystem have the same design capability.

11.3.4 Subsystem TID Capability

A spacecraft consists of subsystems which contain various numbers of electronic circuits, as indicated in a MEL. The approach described in previous sections can now be applied to a subsystem. The assembly of subsystems then leads to the whole spacecraft.

For each subsystem the MEL defines the number of distinct circuits that the subsystem requires for operation. For each circuit within a subsystem (and its redundant counterpart, when applicable) it is possible to generate the hardness probability distributions according to Sections 11.3.1, 11.3.2. Each probability distribution is scaled by its α factor, according to the discussion in Section 11.3.3, to express the TID capability. Each *string* of a subsystem that consists of m boards can be expressed symbolically by the function set $\{P_1(. > d), P_2(. > d), ..., P_m(. > d)\}$, with each P_j being of a different functional form –depending on its constituent components. A random draw from these distributions is made, the result of the draw is a list $l = \{d_1, d_2, \ldots, d_m\}$, where each d_j corresponds to a dosage that just exceeds the hardness of the circuit. The notion that a subsystem will fail when one of its circuits fails means that for the random realization corresponding to list l, the external dosage level at which the subsystem fails is given by the *smallest* number in list l. Thus for a subsystem consisting of m boards the dosage level at failure is

$$d_{SS} = \min\{d_1 \in P_1(. > d), d_2 \in P_2(. > d), ..., d_m \in P_m(. > d)\} \tag{11.14}$$

As seen in (11.5) and (11.10), the probability distributions P_j in (11.14) are already complex and the chances for analytical solutions would be slim, except for very simple but unrealistic distributions. The approach for computing the probability distribution for d_{SS} of necessity needs to be a numeric one. When there are

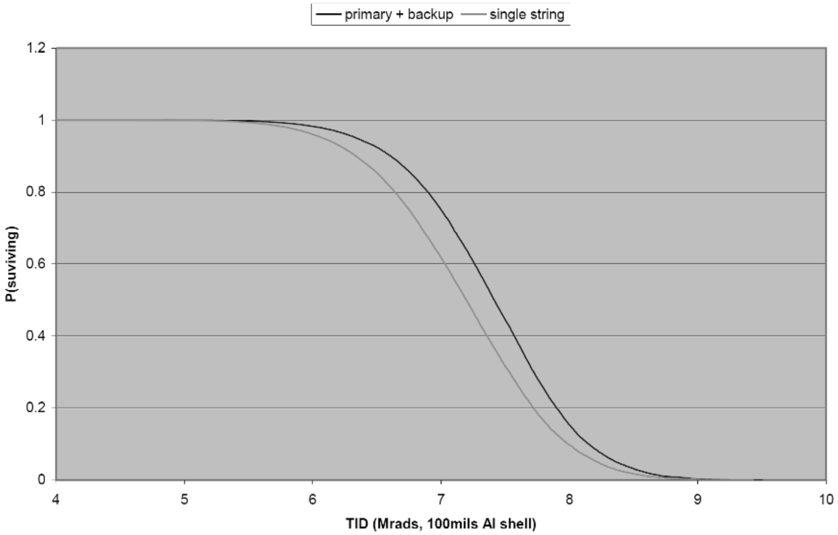


Fig. 11.8: Survival probability for a single string of a notional ACS subsystem compared to that of a primary plus a redundant string

dual strings, a primary as well as a backup subsystem, the dosage capability of the subsystem as a whole becomes

$$d_{SS} = \max(d_{SS}^{primary}, d_{SS}^{backup}) \tag{11.15}$$

where $d_{SS}^{primary}$ and d_{SS}^{backup} represent the dosage capability of the primary and backup. As might be expected the dosage capability for the redundant system should have a higher failure dosage level. An example of the resulting TID capability for the notional Attitude Control Subsystem (ACS) is shown in Fig. 11.8, where the single string TID capability is compared to the dual string case.

For some subsystems the definition of TID capability becomes more complex. For example in many flight system designs there are typically four Reaction Wheel Assembly (RWA) circuits in the ACS subsystem. For a mission to operate nominally only 3 out of 4 are needed. Therefore the TID capability of the RWA component of the ACS subsystem is defined by the second order-statistic of the set $\{d_1, d_2, d_3, d_4\}$. It should be emphasized that since the individual samples are by construction correlated in some non-trivial fashion, the problem is unmanageable analytically. In Fig. 11.9 the TID capability of a single RWA circuit compared to three out of four is shown for the notional circuits. The figure shows the intuitive expectation that for low TID levels the capability of three out of four is greater than an individual one, but as the TID level increases the probability that at least one circuit survives is greater than requiring that at least three survive.

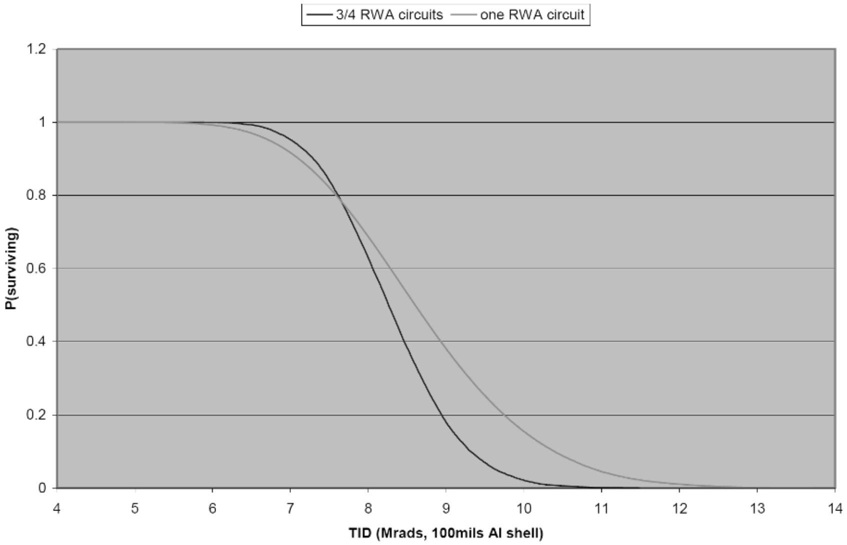


Fig. 11.9: At low TID levels the subsystem survival is greater with three out of four circuits in comparison with a single circuit. At higher TID levels the reverse happens

As another example, in the case of the Command and Data Handling (C&DH) non-volatile memory (NVM), for it to be operational (albeit at reduced storage capability) the model defines the TID capability of the SSR NVM as being the highest TID capability among four circuits, $\max\{d_1, d_2, d_3, d_4\}$.

11.3.5 Flight System and Payload TID Capability

The method to determine the individual TID capability of each subsystem was described in Section 11.3.4. The full flight system is composed of a number of subsystems as categorized in the MEL, Fig. 11.2. In Section 11.3.4 the method for deriving the probability distribution for a subsystem TID capability was described (11.14) or (11.15). The purpose of the flight system is to carry a science payload, as an illustration of a spacecraft with an instrument the model combines the notional flight system and a notional camera payload that consists of camera electronics circuit and three imaging arrays. The pattern of deriving probability distributions in (11.5), (11.10), (11.14), (11.15) leads to a similar characterization. The flight system is divided into ACS, C&DH, Power, Telecom and Propulsion subsystems. For redundancy architecture a cross-strapping (one method to fully utilize all of the redundant components in a system) of the flight subsystems is assumed. Each subsystem j is represented by a TID capability probability distribution $P_{SS_j}(d_j)$.

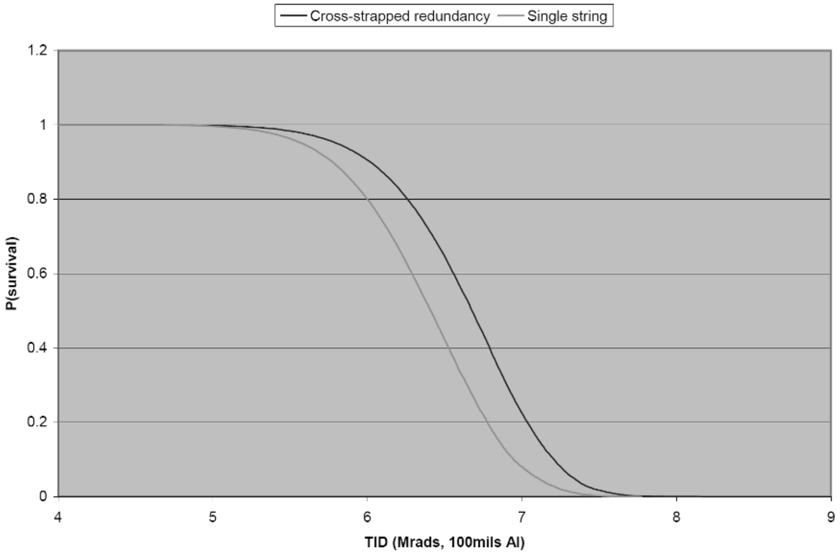


Fig. 11.10: Probability of achieving a TID capability for the full system. A comparison is made between single string and cross-strapped redundant architectures

The complete system is symbolically denoted as

$$\left\{ \begin{array}{c} P_{SS_1}(. > d_1), P_{SS_2}(. > d_2), P_{SS_3}(. > d_3), P_{SS_4}(. > d_4), \\ P_{SS_5}(. > d_5), P_{SS_6}(. > d_6) \end{array} \right\} \tag{11.16}$$

where the indices 1 through 6 refer to subsystems ACS, C&DH, Power, Telecom, Propulsion, and the camera payload. Using (11.14) or (11.15) the model generates random realizations of the complete system which we represent as $\{d_1, d_2, d_3, d_4, d_5, d_6\}$. The full system fails when the smallest TID capability in the set is exceeded by the environment. The model defines the complete system TID capability by

$$\begin{aligned} &D = \min\{d_1 \in P_{SS_1}(. > d_1), d_2 \in P_{SS_2}(. > d_2), d_3 \in P_{SS_3}(. > d_3), \\ &d_4 \in P_{SS_4}(. > d_4), d_5 \in P_{SS_5}(. > d_5), d_6 \in P_{SS_6}(. > d_6)\} \end{aligned} \tag{11.17}$$

For illustration the notional complete system probability distribution of TID capability is shown in Fig. 11.10 for two architectures, a single string design and a cross-strapped redundant design —the benefits of a cross-strapped design are clearly observed.

The discussions in Section 11.3 have been geared towards describing the probabilistic radiation tolerance of a spacecraft using arguments that encapsulate empirical parts testing results, expert opinion, engineering design processes and potential correlations due to parts manufacturing or parts selection. To address the question of the *lifetime* of such a spacecraft the environment to which the spacecraft will be exposed needs to be characterized. This is an area where information is sparse and *epistemic* uncertainties are abundant. An approach for characterizing the "known unknown" needs to be devised, Section 11.4 addresses this point.

11.4 Characterization of the Radiation Environment and *Epistemic* Uncertainties

The radiation environment by nature has uncertainties; uncertainties are due to the physical variability of the environment as well as the *epistemic* uncertainties due to the incompleteness of physics models, as well as sparse and incomplete data. Radiation damage to the spacecraft components is caused by electrons, protons, gamma rays, heavy ions, galactic cosmic rays. The study of [7] indicates that particles of concern for the TID around Europa are high energy electrons. This does not address the non-TID damage caused by radiation. The analysis in [7] provides a characterization of the high energy electron flux for a range of Jovian radii.

If for a given time the spatial mean particle flux of energy E could be expressed as a *scalar field* $\Phi(E,x,y,z,t)$, it could safely be assumed that currently the exact form of Φ is not known due to lack of sufficient data and insufficient knowledge of the physics of the problem. For example one such epistemic uncertainty manifests as the somewhat periodic changes in solar activity –for which there is currently no deterministic prediction approach and thus it is treated as an *aleatory* uncertainty modulated by a long term trend which might be predictable in the future if the intricacies of the Sun could be solved. Some of this knowledge gap in radiation environment around Jupiter will be filled in the future years (before the proposed JEO mission) when the Juno spacecraft arrives there a few years from now.

For the model developed here we recognize that the radiation environment from day to day will be correlated due to the regular orbital period of Europa, the orbital resonance with Io which is known to be source of particles, as well as the tilt of Jupiter's magnetic axis that combined with the 10 hour rotational period of Jupiter generates a modulation of the particle fluxes. By putting these observations together it is very reasonable to surmise that if a power spectral density (PSD) of the fluctuations of particle fluxes is obtained that PSD would show prominent peaks at the above mentioned frequencies in addition to a general trend that is indicative of the longer term correlations. Data from the Galileo mission have indications that temporal correlations in the observed electron fluxes exist [13]. To our knowledge no such PSD analysis has so far been performed in the literature so this observation on the PSD is only an educated speculation at this time.

In the following sections we describe a modeling approach that assumes for a given time interval Δ, the electron flux is constant and that over another time interval Δ it is also a constant but of a different value. The important point described here is that in generating such flux values from interval to interval there will be *temporal correlations* between them. First the use of lognormal distribution for generation of random electron fluxes will be described followed by a description of correlated random time series for expected dosages.

11.4.1 Usage of Lognormal Distribution for Radiation Environment

The analysis of data [7] indicates that by defining an electron flux model I_{model} the parameter $r = \log_{10}(I_{data}/I_{\text{model}})$ follows a normal distribution very closely. That is

$$P(r) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{r-\mu}{\sigma}\right)^2} \tag{11.18}$$

with $\mu \sim 0$. The distribution for r can be changed to use natural logarithms (ln). If $y = \log_{10} z$ then $y = \frac{\ln z}{\ln 10}$, as a result r can also be written as

$$r = \frac{\ln\left(\frac{I_{data}}{I_{\text{model}}}\right)}{\ln 10} \tag{11.19}$$

Writing $\tilde{r} = \ln(I_{data}/I_{\text{model}})$ simplifies the probability distribution of $P(r)$ to

$$P(\tilde{r}) = \frac{1}{\ln 10\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{\tilde{r}/\ln 10-\mu}{\sigma}\right)^2} = \frac{1}{\ln 10\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{\tilde{r}-\mu\ln 10}{\sigma\ln 10}\right)^2} \tag{11.20}$$

Further defining $\tilde{\sigma} = \sigma \ln 10$ we can express the lognormal distribution (11.20) for the observed data with a standard deviation $\tilde{\sigma}$.

We now assume that the observed value of electron flux, I, can be taken to follow a random distribution of the form

$$\frac{1}{I\sqrt{2\pi}\tilde{\sigma}} e^{-\frac{1}{2}\left(\frac{\ln I - \ln I_{\text{model}}}{\tilde{\sigma}}\right)^2} \tag{11.21}$$

At $R_J \sim 9.5$ the results of [7] can be used to show that $\tilde{\sigma} \approx \log_{10}(1.5)\ln 10$.

Assuming that electron flux for a given time interval is a constant, we integrate the flux over the time interval, the result will be proportional to the TID for that time interval. With these assumptions we arrive at the conclusion that the TID, denoted as d, over a finite interval Δ, follows the lognormal distribution also, with a probability distribution

$$P(d) = \frac{1}{d\sqrt{2\pi}\tilde{\sigma}} e^{-\frac{1}{2}\left(\frac{\ln d - \ln d_0}{\tilde{\sigma}}\right)^2} \tag{11.22}$$

The reference radiation design point is 1.65 Mrads at EOI and 2.9 Mrads 105 days later. These values correspond to a mean TID per day of 11.9Krads/day. Based on the science observing scenarios and tour planning most of the radiation dose prior to EOI has been accumulated in the periods just before EOI. For modeling purposes it can be assumed that for most of the mission duration the radiation dosage is small and it begins to accumulate rapidly some time before EOI and rapid accumulation continues past EOI. Using this simplifying assumption the zero point of dosage can be found by marching back in time at the same dose level as in post EOI. By this method it is found that ˜140 days prior to EOI corresponds to the zero point of dosage.

11.4.2 Methodology for Generating Correlated Ionizing Dose Time Series

To begin the generation of dosage time series starting from EOI-140d we can consider accumulated dosage on a time interval Δ of 15 days as the building block for generating the time series; this also allows for speedy calculations. To create correlated time series using intervals shorter than 15 days will make the computations more time consuming without adding substantive information because of the inherent uncertainties in the radiation models.

With smoothed dosage increments that were described in Section 11.4.1 we use two accumulation scenarios for pre-EOI and post-EOI periods. For periods prior to EOI the assumption of uncorrelated 15-day increments may be justifiable because the spacecraft travels over a wide range of distances and environments. In the post-EOI period the assumption of an uncorrelated behavior between dosage increments is not a reasonable one for the reasons discussed in Section 11.4. For the post-EOI period it is necessary to generate random data that are *correlated.*

Due to the lack of spectral analysis of the radiation environment as discussed in Section 11.4, it becomes necessary to adopt some model of the temporal behavior of encountered dosages. We will adopt a simple *correlation function* for the increments, with a single parameter that allows varying the degree of temporal correlation, from uncorrelated to fully correlated. The correlation function used here is of the form $\exp(-|t-t'|/\tau)$, with t and t' referring to the times corresponding to each 15-day increment, and τ is the time constant of the temporal correlation.

Time is discretized into a sequence $\{0,1,2,\ldots,n\}$ where 0 refers to time at the dosage zero point —which we approximate to be EOI-150 days (to make it divisible by 15 for simplicity). The end point of the sequence is taken to be a long time after EOI. For computational purposes we take the end point to be EOI+540 days. The time period EOI-150days to EOI+540days is divided into 15-day intervals. For each 15 day interval a mean dosage accumulation of 15*11.9Krads =.179Mrads is adopted.

For the individual dosage accumulations over each time interval denoted by $\{0,1,2,\ldots,n\}$ a random data set $\{\delta_0,\delta_1,\delta_2,...,\delta_n\}$ is created. The time series for the *total* accumulation over time is given by the sequence $\{\xi_0, \xi_1, \xi_2, ..., \xi_n\}$ where $\xi_i = \xi_{i-1} + \delta_i$. At the end of the zeroth dose time interval $\xi_0 = \delta_0$.

Since all δ_i are positive, the generated sequence is a monotonically increasing stochastic process.

For the first 11 samples –corresponding to pre-EOI period, the stochastic chain acts like a random walk, but with positive increments. Each increment is taken to be an independent random draw from a lognormal distribution with mean value of 0.179Mrads and a standard deviation of $\log_{10}(1.5)\ln 10 \approx 0.405$. For the remaining increments starting from EOI, each one is still drawn from a lognormal distribution with same mean and standard deviation as before EOI, but now the *increments are forced to be correlated.* With the correlation function that was defined earlier, the random number generation process must satisfy the following correlation matrix ρ, this matrix is symmetric positive definite with the following structure, where $\rho_{ij} = \exp(-\left|t_i - t_j\right|/\tau)$.

$$\rho = \begin{pmatrix} 1 & \rho_{12} & \rho_{13} & \cdot & \cdot & \cdot & \rho_{1n} \\ \rho_{12} & 1 & \rho_{23} & \cdot & \cdot & \cdot & \cdot \\ \rho_{13} & \rho_{23} & 1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \rho_{1n-1} & \cdot & \cdot & \cdot & \cdot & 1 & \rho_{n-1n} \\ \rho_{1n} & \rho_{12} & \cdot & \cdot & \rho_{n-2n} & \rho_{n-1n} & 1 \end{pmatrix} \tag{11.23}$$

The calculations in the model extend to EOI+540days, so ρ in (11.23) is a 36×36 matrix. An example for several sequences generated according to this recipe is shown in Fig. 11.11 for the case of complete correlation between the increments (this is the case when the time constant τ is infinite).

Another example when the temporal correlations have smaller time constants (of the order of many weeks) is shown in Fig. 11.12. As a side remark we note that the lognormal distribution is a "fat-tailed" distribution, as a consequence a random walk that uses a lognormal distribution for its increments will display sudden jumps, as this figure shows. As the increment to increment correlation increases the dosage time histories become more like straight lines as in Fig. 11.11.

11.5 Estimation of the System Lifetime

The method of estimating system TID capability was described in Section 11.3.5 and the method for generating random dosage time histories was developed in Section 11.4.2. By appropriate combination of these two results the lifetime probability distribution can be derived. Symbolically suppose a dosage history (in reference to 100mils Al) is written as $D(t)$. Examples of such series are shown in Figs. 11.11 and 11.12. An assumption is also adopted that the dosage at all circuits is the same for any given time. This is the assumption of *complete spatial correlation* in the radiation environment *on the scale of the spacecraft.* If in the future it becomes possible

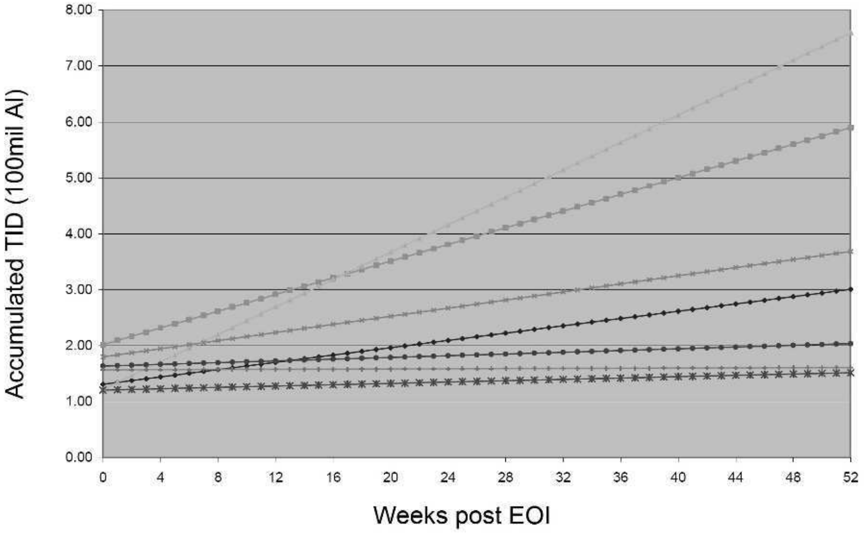


Fig. 11.11: A few realizations of total ionizing dose time series when the individual temporal increments are assumed to be completely correlated

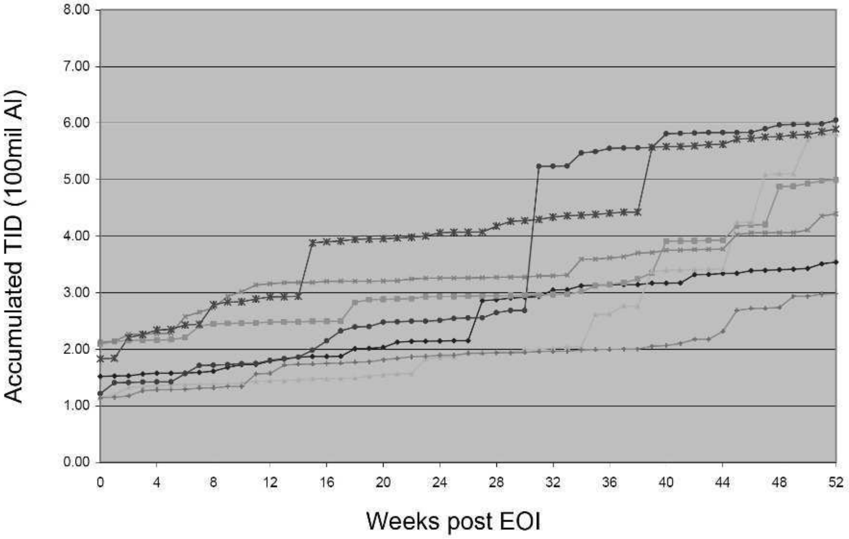


Fig. 11.12: Several realizations of dosage time series when the individual increments are correlated with a time correlation constant of the order of many weeks

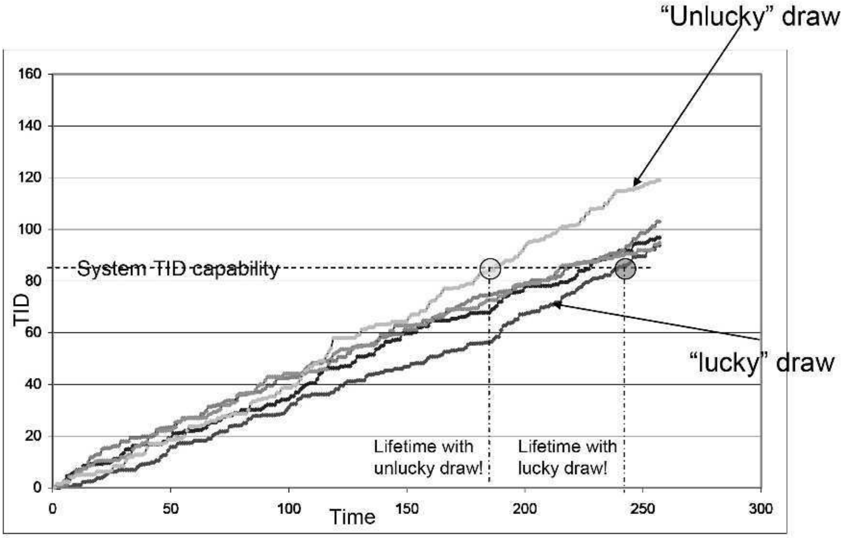


Fig. 11.13: For a given system TID capability the "roll of dice" represented by the random radiation time series determines the lifetime of the spacecraft

to determine how the radiation environment varies over the scale of the spacecraft, the model can be straightforwardly modified to generate multiple dosage histories for each realization of the dosage history –where all realizations are very close to the mean dosage $D(t)$ but differ sufficiently from each other to represent spatial variations due to spacecraft self shielding or directionality in the radiation field. For simplicity this model uses one single dosage history for the whole system.

Setting aside all spatial variations in the radiation environment at the location of the spacecraft, the scaled capabilities of each circuit are in reference to the same single environment. It is therefore possible to use a *single* dosage realization for estimating the system lifetime. Suppose a random draw from the system TID capability probability distribution (11.17) is made and also a random dosage history is generated. For this realization of the system + environment the lifetime of the system is at the intersection of the system TID capability and the dosage time series. By performing many realizations a probability distribution function for the system lifetime can be generated. A graphical example of how this works is shown in Fig. 11.13. In this figure if the radiation dosage time series happens to be benign the lifetime will be longer than if the dosage time series had been more unforgiving. Some realizations are lucky and some are not! If a system has a TID capability as shown, the lifetime depends on the radiation scenario. For the same TID capability the unlucky draw gives a shorter lifetime than a lucky draw would.

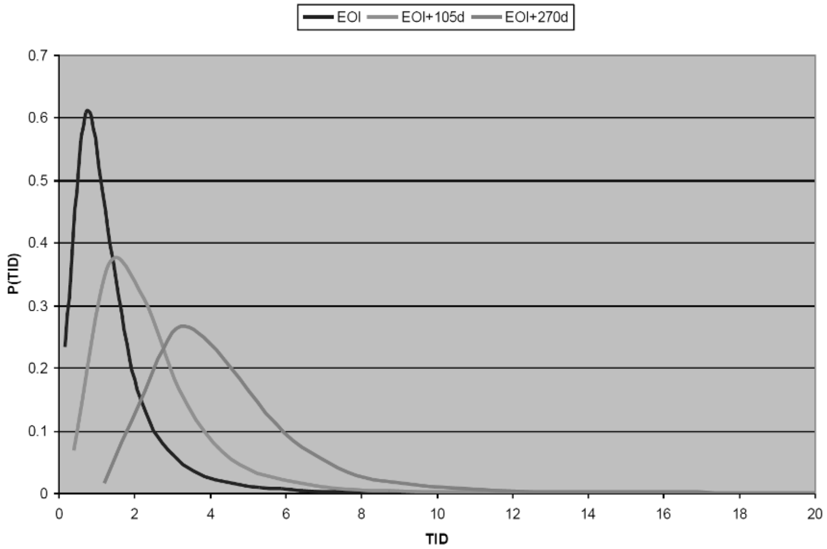


Fig. 11.14: Probability distribution of the encountered TIDs at various epochs when the dosage increments are uncorrelated

11.6 Lifetime Probabilities and Effects of Temporal Correlations for a few Scenarios

Using the approach for estimating the system lifetime in Section 11.4.2 we can probe various scenarios for the assumed electronic component hardness characteristics (μ, SF and C_{OV} shown in Table 11.1) as well as the degree of temporal correlation in the radiation environment. For the component characteristics we use a sampling of the values shown in Table 11.1 and assess the lifetime for three radiation correlation assumptions, one with zero correlation from one increment to the next, one with a time constant of 45 days and another with a time constant of 90 days. Assuming a radiation history that is non-correlated (which is usually the easiest way most analysis is done) the probability distribution of the encountered TIDs at various epochs is shown in Fig. 11.14. The resulting lifetime distribution for the same set of assumptions is shown in Fig. 11.15.

When the time correlation constant is increased to 45 days the probability distribution of the encountered TIDs widens with the extremes, both high and low becoming more likely. The means of the distributions remain unchanged because of the use of the same mean value for every increment. Figure 11.16 illustrates the widening of the encountered TID probability distributions due to the increased time correlation constant. The system lifetime probability in this case reflects the widening of the encountered TID probability distribution. The probability of survival in the early phases of the mission is reduced because the scenario allows higher prob-

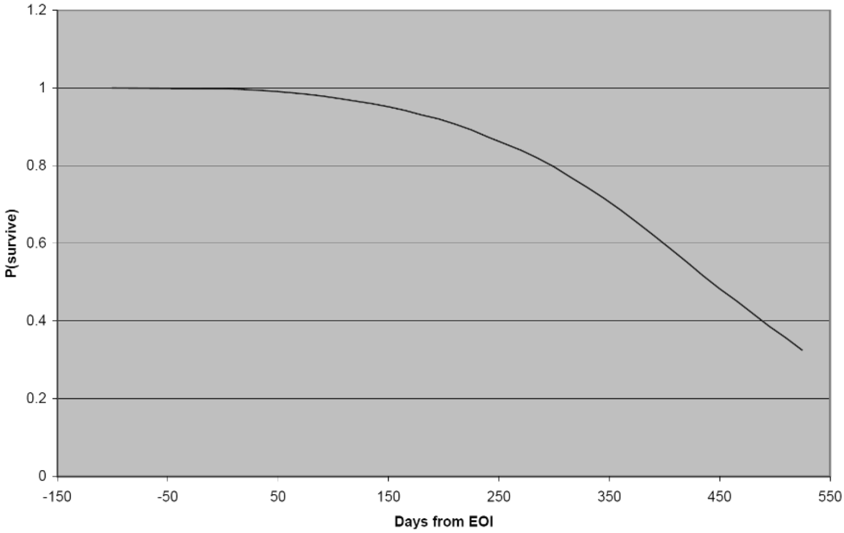


Fig. 11.15: Probability of system survival assuming a temporally uncorrelated radiation environment

abilities for large TIDs; the increased probability for longer lifetimes is because the scenario also allows more low TIDs as well; this effect is seen in Fig. 11.17 for the cases of $\tau = 0$, 45 and 90 days.

11.7 Conclusions

The results described here have extended the previous lifetime modeling effort that used simple counting of parts and considered only two extreme cases of radiation environment, either fully uncorrelated or fully correlated day to day radiation dosages. In this work by using a description of the flight system based on the MEL and developing a new approach to characterize the inherent *epistemic* uncertainty in the environment at Europa we have enabled assessment of additional hardware architectures and hardness scenarios as well as varying the assumptions for the effects of the radiation environment on the flight system.

The problem discussed here is representative of many other situations where making probabilistic predictions becomes entangled in a web of *epistemic* uncertainty and sparseness of information. We sketched an approach for dividing the larger problem into manageable portions where quantification of probabilities can be attempted. Even in smaller scale type of problems the issues of *epistemic* uncertainties and use of expert opinion needs to be addressed. The approach described here could be adaptable to many other cases. While the experts' opinion or the un-

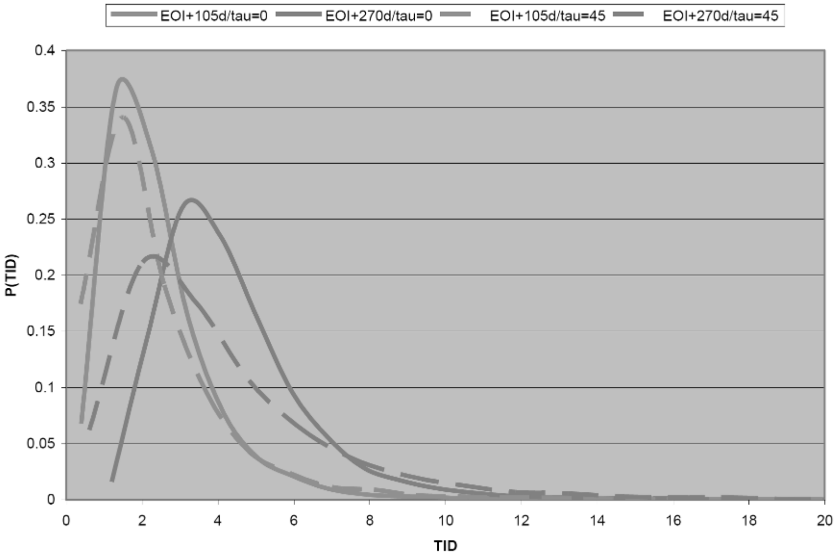


Fig. 11.16: Widening of the probability distribution of TIDs due to a time correlation constant of 45 days

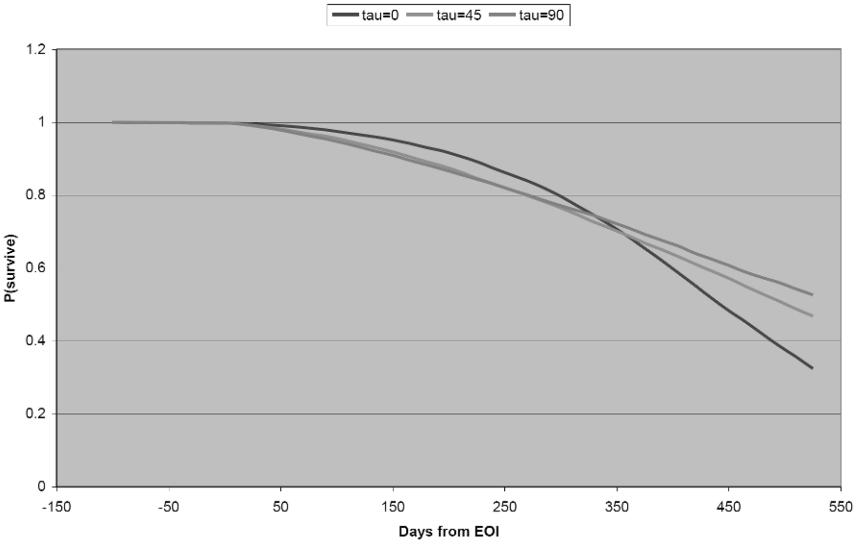


Fig. 11.17: Changes in system lifetime as the time correlation constant of the radiation environment is increased

known environment might be of a different nature, a characterization of what is "known to be unknown" can be made feasible. A simple model of what is "known to be unknown" is preferable to *ad hoc* approaches that can not be parameterized in a simple fashion. The conclusions from this analysis, using more representative values of the parameters than were shown here have been leveraged to lend concrete support to the concept that a mission to a high radiation environment such as Europa is not in the realm of science fiction. Such a mission to Europa may become possible by the end of this decade.

Acknowledgements The evolution of this work has benefited from fruitful discussions with Bob Rasmussen, Chet Everline, Allan Johnston, Ed Shalom, and the late Guy Man. The research described here was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

1. P. D. Fiesler, S. M. Ardalan, A. R. Frederickson, "The Radiation Effects on Galileo Spacecraft Systems at Jupiter", *IEEE Transactions on Nuclear Science*, **49**, 6, 2739-2758, 2002.
2. B. E. Pritchard, G. M. Swift, A. H. Johnston, "Radiation Effects Predicted, Observed, and Compared for Spacecraft Systems", *2002 IEEE Radiation Effects Data Workshop*, 7-13, 2002.
3. M. Tafazoli, "A study of on-orbit spacecraft failures", *Acta Astronautica*, **64**, 195-205,2009.
4. D.J. Sheldon, "Electronic Failures in Spacecraft Environments", *Reliability Physics Symposium (IRPS), 2010 IEEE International*, 759-762, 2010.
5. Design, Verification/Validation & Ops Principles for Flight Systems (Design Principles), Rev. 4, *JPL Rules* DocID 43913, 2010.
6. 2010 Joint Jupiter Science Definition Team Report to NASA, JPL D-67959, 2010.
7. I. Jun, H. B. Garrett, R. Swimm, R. W. Evans, G. Clough, "Statistics of the variations of the high-energy electron population between 7 and 28 jovian radii as measured by the Galileo spacecraft", *ICARUS*, **178**, 386-394, 2005.
8. C. F. Guenther, "A method to quantitatively justify and relate shielding requirements and design margins to hardware requirements", *Digital Avionics Systems Conference Proceedings, IEEE/AIAA/NASA 9th*, 425-429, 1990.
9. Charles D. Brown, Elements of Spacecraft Design, American Institute of Aeronautics and Astronautics, Inc., 2002.
10. Palisade Corporation, @Risk for Excel, http://www.palisade.com/.
11. C. Everline, K. Clark, G. Man, R. Rasmussen, A. Johnston, C. Kohlhase, T. Paulo,"Estimating the Reliability of Electronic Parts in High Radiation Fields", *Proceedings of the International Conference on Probabilistic Safety Assessment and Management, PSAM 9*, 2008.
12. M. Moshir, D. W. Murphy, D. L. Meier, M. H. Milman, "Systems engineering and application of system performance modeling in SIM Lite Mission", Proc. of SPIE, Vol. 7734, 77341H, 2010.
13. Private communication, H. B. Garrett.
14. A. H. Johnston, Reliability and Radiation Effects in Compound Semiconductors, World Scientific Publishing Company, 2010.
15. H. A. David, H. N. Nagarja, Order Statistics, Wiley-Interscience, 2003.

Chapter 12
Robust State and Parameter Estimation for Nonlinear Continuous-Time Systems in a Set-Membership Context

Denis Efimov, Tarek Raïssi (✉), and Ali Zolghadri

Abstract This chapter deals with joint state and parameter estimation for nonlinear continuous-time systems. Based on an appropriate LPV approximation, the problem is formulated in terms of a set adaptive observer design problem which can be efficiently solved. The resolution methodology avoids the exponential complexity obstruction often met in set-membership parameter estimation. The efficacy of the proposed set adaptive observers is demonstrated on several examples.

12.1 Introduction

Observer design for nonlinear systems has been an area of intensive research during the last two decades. There exist a lot of solutions dealing with diverse forms of system models, see for instance [1, 2]. Typically, the observer design problem is solvable if the system model can be transformed to a canonical form, that may be an unacceptable assumption in many applications. Consider a generic nonlinear system

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \theta, \mathbf{d}), \mathbf{y} = \mathbf{h}(\mathbf{x}) + \mathbf{v} \tag{12.1}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, $\theta \in \mathbb{R}^q$, $\mathbf{d} \in \mathbb{R}^l$, $\mathbf{y} \in \mathbb{R}^p$, $\mathbf{v} \in \mathbb{R}^p$ are respectively the state, the control, the unknown parameters, the disturbances, the output and the measurement

Denis Efimov
IMS-lab, University of Bordeaux, 351 cours de la libération, 33405 Talence, France
e-mail: denis.efimov@ims-bordeaux.fr

Tarek Raïssi
IMS-lab, University of Bordeaux, 351 cours de la libération, 33405 Talence, France
e-mail: tarek.raissi@ims-bordeaux.fr

Ali Zolghadri
IMS-lab, University of Bordeaux, 351 cours de la libération, 33405 Talence, France
e-mail: ali.zolghadri@ims-bordeaux.fr

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_12,

noise; $t \in \mathbb{R}_+$. The functions $\mathbf{f}$, $\mathbf{h}$ are continuous with respect to all arguments and differentiable with respect to $\mathbf{x}$ and $\mathbf{u}$. In the literature, several observers are built based on an approximation (or a transformation) of the nonlinear model (12.1) to a Linear Parametric-Varying (LPV) one [3, 11]. LPV models are described by:

$$\dot{\mathbf{x}} = \mathbf{A}(\rho(t))\mathbf{x} + \mathbf{B}(\rho(t))\mathbf{u}, \mathbf{y} = \mathbf{C}(\rho(t))\mathbf{x} + \mathbf{v} \tag{12.2}$$

where the scheduling parameter vector $\rho \in \mathscr{P}$ is a priori unknown, but with known bounds, and $\mathscr{P}$ is a set of functions that remain in a compact real subspace. When ρ is a function of the system state, the model (12.2) is called quasi-LPV. Furthermore, it is important to note that the system (12.2) is an equivalent representation of (12.1), in the sense that trajectories of (12.1) remain in the trajectories of (12.2). Among the available methodologies for LPV model constructions one can mention the Jacobian linearization, the state transformation and the state substitution approaches [4, 12, 13]. The idea is to replace nonlinear complexity of the model (12.1) by enlarged parametric variation in the linear model (12.2). Such LPV transformation simplifies the design of an observer for the system (12.1). As it will be shown in the sequel, sometimes the complete LPV linearization is not necessary and a partial one may be more suitable. For example, for the observer design purposes some nonlinearities depending only on the output $\mathbf{y}$ can be preserved in order to decrease the uncertainties of the model (12.2) collected in the vector ρ. The observer design methodology proposed in the sequel is based on a guaranteed LPV transformation recently developed in [14]. By "guaranteed", it is understood that the nonlinear trajectory is sure to remain in the set of trajectories of the resulting LPV model. It is based on an interval linearization around the operational state domain instead of a linearization throughout the equilibrium points. The proposed LPV approximation is performed by means of interval analysis [5, 6].

In the following, an adaptive set observer is developed based on (12.2) in a set-membership context. There exist three main approaches to perform interval state estimation for systems described by (12.2): the prediction/correction mechanism as in the Kalman filter [15, 16]; the approach based on comparison theorems [17, 18]; and the closed loop interval observers with cooperative observation error dynamics [19–21]. The latter has been extended in [14] for nonlinear systems using LPV approximations with known minorant and majorant matrices for (12.2). Unfortunately, these state estimators are efficient only when the uncertainties of the parameter vector are not large.

To the best of our knowledge, joint state and parameter estimation has not been fully studied for systems described by (12.1) in a bounded error context and only some attempts have been made to solve this problem [10, 16, 23]. In [16], the parameter estimation problem is formulated as a set inversion and solved by the SIVIA algorithm (Set Inversion Via Interval Analysis) [22]. An inclusion test involving a validated integration of a set of ordinary differential equations (ODEs) should be evaluated over a time horizon. Such a procedure is computationally time-consuming since the complexity of SIVIA is exponential with respect to the parameter vector dimension. In [23] the validated integration of ODEs is associated with consistency techniques in order to reduce the computing time. Nevertheless, the algorithm in [23] is efficient only for very moderate levels of noise and the complexity remains exponential. In the following, the methodology proposed in [14] is extended to deal with joint state and parameter estimation even for high dimensional systems and with large parametric uncertainties. The idea is to develop set-membership adaptive observers based on the works reported in [24–26, 29].

In this work a procedure for adaptive set observer design is proposed for a subclass of the LPV representation (12.2). The main feature of this step is that cooperativity property of the state observers (which can be assigned by the proper choice of the observer gain, see [14] for more details) is not inherited by the adaptive counterpart. Resolution of this issue requires especial consideration and additional conditions checking. The main advantage is that no bisection is needed in the parameter estimation procedure and the complexity of the algorithm is not exponential. Thus, such an observer could be used even for high dimensional systems.

This chapter is organized as follows. In Section 12.2 the formal problem statement is presented. Some preliminaries are given in Section 12.3. The adaptive observer equations and the applicability conditions for the adaptive set observer are derived in Section 12.4. Two different sets of conditions are analyzed leading to cooperative or competitive adaptive observer loops. The combined set state observer is analyzed in Section 12.5. Through the chapter numerical examples are provided to illustrate the proposed results.

12.2 Problem Statement

Let us assume that the system (12.1) can be transformed to the following form:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}(\rho(t))\mathbf{x} + \mathbf{B}(\rho(t))\mathbf{u} + \varphi(\mathbf{y}) + \mathbf{G}(\mathbf{y})\theta \\ \mathbf{y} = \mathbf{C}\mathbf{x}, \mathbf{y}_v = \mathbf{y} + \mathbf{v} \end{cases}, \tag{12.3}$$

where $\mathbf{x} \in X \subset \mathbb{R}^n$, $\mathbf{u} \in U \subset \mathbb{R}^m$, $\mathbf{y} \in Y \subset \mathbb{R}^p$ are the state, the input and the output vectors; $\theta \in \Theta \subset \mathbb{R}^q$ is the vector of uncertain parameters; $\mathbf{v} \in V \subset \mathbb{R}^p$ is the measurement noise; $\mathbf{y}_v$ is the vector of noisy measurements of the system (12.3), $\rho \in \Upsilon \subset \mathbb{R}^r$ is some scheduling parameter vector. The compact sets X, U, Y, V, Θ and Υ are given *a priori* and it is assumed that there exist some constant vectors $\mathbf{x}_m, \mathbf{x}_M \in R^n$ such that $\mathbf{x}_m \leq \mathbf{x} \leq \mathbf{x}_M$ for all $\mathbf{x} \in X$. The vector function φ and columns of the matrix function $\mathbf{G}$ are locally Lipschitz continuous and $\mathbf{C}$ is a constant matrix of appropriate dimensions. The majorant matrices $\mathbf{A}_m$, $\mathbf{A}_M$, $\mathbf{B}_m$, $\mathbf{B}_M$ are given such that

$$\mathbf{A}_m \prec \mathbf{A}(\rho) \prec \mathbf{A}_M, \mathbf{B}_m \prec \mathbf{B}(\rho) \prec \mathbf{B}_M$$

for all $\rho \in \Upsilon$ (the inequality $\mathbf{A} \prec \mathbf{B}$ for matrices $\mathbf{A}$, $\mathbf{B}$ with dimension $n \times m$ is understood elementwise $A_{i,j} \leq B_{i,j}$, $i = \overline{1,n}$, $j = \overline{1,m}$). Note that, since $\mathbf{y} \in Y$ and $\mathbf{v} \in V$ there exist constants $k_\varphi > 0$, $k_G > 0$ such that $|\varphi(\mathbf{y}) - \varphi(\mathbf{y}_v)| \leq k_\varphi |\mathbf{v}|$ and $|\mathbf{G}(\mathbf{y}) - \mathbf{G}(\mathbf{y}_v)| \leq k_G |\mathbf{v}|$.

Remark 12.1. Note that the output map in (12.3) is linear and the dynamic effect of the output and uncertain parameters is modeled through the term $\mathbf{G}(\mathbf{y})\,\theta$. Obviously, this model structure introduces some restrictions on the system (12.1) and its LPV transformation. In addition, it is assumed that in the system (12.3), the LPV transformation is not applied to some nonlinear terms dependent only on the output $\mathbf{y}$ and, the functions φ and $\mathbf{G}$ are preserved in their original form. In fact, to increase accuracy of the system (12.1) LPV approximation, one should explicitly handle with care the output dependency in all nonlinearities, thus the most accurate presentation of (12.3) could be

$$\dot{\mathbf{x}} = \mathbf{A}(\rho(t),\mathbf{y})\,\mathbf{x} + \mathbf{B}(\rho(t),\mathbf{y})\,\mathbf{u} + \varphi(\mathbf{y}) + \mathbf{G}(\mathbf{y})\,\theta.$$

In some examples below we will consider this issue with more details, however for brevity of presentation, all theoretical results will be formulated only for the system (12.3) (an extension on the former case is trivial).

In the following the aim is to design an adaptive observer that, in the noise-free case, provides interval observation of unmeasured components of the state vector $\mathbf{x}$ in (12.1) and estimates the set of admissible values for the vector θ. For any $\mathbf{v}(t) \in V$, $t \geq 0$ the observer solutions should be bounded.

Note that the model form (12.3) is very important in several engineering fields such as fault detection [28]. In this case, the vector θ could be composed of two parts: the first one represents the physical parameters which are not exactly known and the second part contains some "fictive" parameters used to model the effect of faults. The latter parameters (or some of them) become significantly different from their nominal range when a fault occurs. Without loss of generality, the fictive parameters are assumed to have zero value in the nominal fault free case. Moreover, detecting and isolating parametric faults require that the system is sufficiently excited that motivates the requirement of persistent excitation for time varying dynamical systems recalled in the following section.

12.3 Preliminaries

12.3.1 Monotone Systems

The system

$$\dot{\mathbf{x}} = \mathbf{f}(t,\mathbf{x}), \mathbf{x} \in X, t \geq 0 \tag{12.4}$$

with the solution $\mathbf{x}(t,\mathbf{x}_0)$ for the initial condition $\mathbf{x}(0) = \mathbf{x}_0$ is called monotone, if $\mathbf{x}_0 \leq \xi_0 \Rightarrow \mathbf{x}(t,\mathbf{x}_0) \leq \mathbf{x}(t,\xi_0)$ for all $t \geq 0$ [7] (for the vectors $\mathbf{x}_0$, ξ_0 the inequality $\mathbf{x}_0 \leq \xi_0$ is understood elementwise). The system (12.4) is called cooperative if $\partial f_i(t,\mathbf{x})/\partial x_j \geq 0$ for all $1 \leq i \neq j \leq n$, $t \in \mathbb{R}$ and $\mathbf{x} \in X$ [7]. Cooperative systems form a subclass of monotone ones. A matrix $\mathbf{A}$ with dimension $n \times n$ is called cooperative if $A_{i,j} \geq 0$ for all $1 \leq i \neq j \leq n$. Note that for the cooperative stable system (the matrix $\mathbf{A}$ is cooperative and Hurwitz)

$$\dot{\mathbf{s}}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{r}(t), \quad \mathbf{s} \in \mathbb{R}^n, \mathbf{r} \in \mathbb{R}^n, t \geq 0$$

the properties $\mathbf{s}(0) \geq 0$, $\mathbf{r}(t) \geq 0$ for all $t \geq 0$ imply $\mathbf{s}(t) \geq 0$ for $t \geq 0$ and, conversely, $\mathbf{s}(0) \leq 0$, $\mathbf{r}(t) \leq 0$ for all $t \geq 0$ ensures $\mathbf{s}(t) \leq 0$ for $t \geq 0$. The system (12.4) is called competitive if $\partial f_i(t,\mathbf{x})/\partial x_j \leq 0$ for all $1 \leq i \neq j \leq n$, $t \in \mathbb{R}$ and $\mathbf{x} \in X$, the competitive systems behave like cooperative in backward time [7].

12.3.2 Persistency of Excitation

The (Lebesgue) measurable and square integrable matrix function $\mathbf{R} : \mathbb{R} \to \mathbb{R}^{l_1 \times l_2}$ with dimension $l_1 \times l_2$ admits a (ℓ, ϑ)-persistency of excitation (PE) condition, if there exist strictly positive constants ℓ and ϑ such that

$$\int_t^{t+\ell} \mathbf{R}(s)\mathbf{R}(s)^T\, ds \geq \vartheta \mathbf{I}_{l_1}$$

for any $t \in \mathbb{R}$, where $\mathbf{I}_{l_1}$ denotes the identity matrix of dimension $l_1 \times l_1$. This property means that the matrix function $\int_t^{t+\ell} \mathbf{R}(s)\mathbf{R}(s)^T\, ds$ has an empty left kernel space on a sufficiently long time interval.

Lemma 12.1. *[24] Consider the time-varying linear dynamical system*

$$\dot{\mathbf{p}} = -\Gamma \mathbf{R}(t)\mathbf{R}(t)^T \mathbf{p} + \mathbf{b}(t), t_0 \in \mathbb{R}_+,$$

where $\mathbf{p} \in \mathbb{R}^{l_1}$, Γ is a positive definite symmetric matrix of dimension $l_1 \times l_1$ and the functions $\mathbf{R} : \mathbb{R}_+ \to \mathbb{R}^{l_1 \times l_2}$, $\mathbf{b} : \mathbb{R}_+ \to \mathbb{R}^{l_1}$ are measurable, $\mathbf{b}$ is essentially bounded, function $\mathbf{R}$ is (ℓ, ϑ)-PE for some $\ell > 0$, $\vartheta > 0$. Then, for any initial condition $\mathbf{p}(t_0) \in \mathbb{R}^{l_1}$, the solution of the system is defined for all $t \geq t_0$ and verifies ($\gamma > 0$ is the smallest eigenvalue of the matrix Γ)

$$|\mathbf{p}(t)| \leq |\mathbf{p}(t_0)| e^{-0.5\gamma\vartheta\ell^{-1}(t-t_0-\ell)} + (1 + 2\vartheta^{-1}\gamma^{-1}e^{-0.5\vartheta\gamma})\ell||\mathbf{b}||.$$

This lemma states that a linear system with a persistently excited time-varying matrix gain and a bounded additive disturbance has bounded solutions.

12.4 Interval Parameters Estimation

To proceed, we would like to introduce the following assumptions dealing with stabilizability by output feedback of the system (12.3) linear part.

Assumption 1. There exist matrices $\mathbf{L}$, $\mathbf{Q}=\mathbf{Q}^T>0$ and $\mathbf{P}=\mathbf{P}^T>0$ such that

$$[\mathbf{A}(\rho)-\mathbf{L}\mathbf{C}]^T\mathbf{P}+\mathbf{P}[\mathbf{A}(\rho)-\mathbf{L}\mathbf{C}]=-\mathbf{Q}$$

for all $\rho\in\Upsilon$.

□

For the system

$$\dot{\mathbf{s}}=[\mathbf{A}(\rho(t))-\mathbf{L}\mathbf{C}]\mathbf{s}+\mathbf{r}, \tag{12.5}$$

$\mathbf{s}\in\mathbb{R}^n$, $\mathbf{r}\in\mathbb{R}^n$, $\rho(t)\in\Upsilon$ for $t\geq 0$, this assumption ensures uniform asymptotic stability property for $\mathbf{r}=0$ and boundedness of the system solutions for any bounded input $\mathbf{r}$ (the input-to-state stability property holds [27]). The system (12.5) is the linear part of (12.3) closed by output feedback with a gain $\mathbf{L}$. This assumption is required for classical adaptive observer design for the system (12.3). It will be shown later that this assumption is not actually required for the proposed approach. It will be relaxed leading to the following assumption, that ensures existence of an adaptive set observer for (12.3).

Assumption 2. There exist gains $\mathbf{L}_m$, $\mathbf{L}_M$ such that the matrices $\mathbf{A}_m-\mathbf{L}_m\mathbf{C}$ and $\mathbf{A}_M-\mathbf{L}_M\mathbf{C}$ are Hurwitz and cooperative, and for all $\mathbf{y}\in Y$, $\mathbf{v}\in V$ we have $0\prec\mathbf{G}(\mathbf{y}+\mathbf{v})$.

□

In addition, since $\theta\in\Theta$, there exist two vectors $\theta_m\in\mathbb{R}^q$ and $\theta_M\in\mathbb{R}^q$ such that $\theta_m\leq\theta\leq\theta_M$ for all $\theta\in\Theta$. Based on these assumptions, the equations of adaptive observer are introduced below in two steps.

12.4.1 Ideal Case

Firstly, assume that the signal $\rho(t)\in\Upsilon$ is available for measurements and assumption 1 holds. Then, an adaptive observer [25,26] for the system (12.3) could be built as:

$$\dot{\zeta}=\mathbf{A}(\rho(t))\zeta+\mathbf{B}(\rho(t))\mathbf{u}+\varphi(\mathbf{y}_v)+\mathbf{L}\,(\mathbf{y}_v-\mathbf{C}\zeta); \tag{12.6}$$

$$\dot{\Omega}=[\mathbf{A}(\rho(t))-\mathbf{L}\mathbf{C}]\,\Omega-\mathbf{G}(\mathbf{y}_v); \tag{12.7}$$

$$\dot{\widehat{\theta}}=-\Gamma_0\Omega^T\mathbf{C}^T\,(\mathbf{y}_v-\mathbf{C}\zeta+\mathbf{C}\Omega\,\widehat{\theta}),\Gamma_0=\Gamma_0^T>0, \tag{12.8}$$

where $\zeta\in\mathbb{R}^n$ is the vector of ”estimates” for $\mathbf{x}$; the matrix $\Omega\in R^{n\times q}$ is an auxiliary variable, which helps to overcome high relative degree obstruction in the system

(12.3), i.e. to identify the value of θ even in the cases when only higher order time derivatives of the output $\mathbf{y}$ depend on θ; $\widehat{\theta} \in R^q$ is the estimate of θ. Defining the observation error $\varepsilon = \mathbf{x} - \zeta$, the estimation error $\tilde{\theta} = \theta - \widehat{\theta}$ and the auxiliary variable $\delta = \varepsilon + \Omega\,\theta$ we obtain

$$\dot{\varepsilon} = [\mathbf{A}(\rho(t)) - \mathbf{L}\,\mathbf{C}]\,\varepsilon + \mathbf{G}(\mathbf{y}_v)\,\theta + \mathbf{d}_v, \tag{12.9}$$

$$\mathbf{d}_v = \varphi(\mathbf{y}) - \varphi(\mathbf{y}_v) + [\mathbf{G}(\mathbf{y}) - \mathbf{G}(\mathbf{y}_v)]\,\theta - \mathbf{L}\,\mathbf{v},$$

$$\dot{\delta} = [\mathbf{A}(\rho(t)) - \mathbf{L}\,\mathbf{C}]\,\delta + \mathbf{d}_v, \tag{12.10}$$

$$\dot{\tilde{\theta}} = \Gamma_0\,\Omega^T \mathbf{C}^T\,(\mathbf{C}\,\delta + \mathbf{v} - \mathbf{C}\,\Omega\,\tilde{\theta}). \tag{12.11}$$

As in [24–26,29], if assumption 1 is satisfied and $\mathbf{y} \in Y$, $\mathbf{v} \in V$, then since the systems (12.7) and (12.10) have a form similar to (12.5), all solutions of the system (12.7) are bounded, i.e. there exists $k_\Omega > 0$ such that $|\Omega(t)| \leq k_\Omega$ for all $t \geq 0$. Furthermore, we have $|\mathbf{d}_v| \leq [k_\varphi + k_G|\theta| + |\mathbf{L}|]|\mathbf{v}|$ for $\theta \in \Theta$, $\mathbf{v} \in V$, then the signal $\mathbf{d}_v$ remains bounded with amplitude proportional to that of $\mathbf{v}$. Therefore, the solutions of (12.10) are bounded and for the case $\mathbf{v}(t) = 0$, $t \geq 0$ the system is asymptotically stable. In addition, if the signal $\Omega^T(t)\,\mathbf{C}^T$ is persistently exciting, then from lemma 12.1 the estimation error $\tilde{\theta}(t)$ remains bounded, and for $\mathbf{v}(t) = 0$, $t \geq 0$ the asymptotic relation holds: $\lim_{t\to+\infty} \widehat{\theta}(t) = \theta$. Finally, $\varepsilon(t) = \delta(t) - \Omega(t)\,\theta$ for all $t \in \mathbb{R}$ and the observation error is bounded since the signals $\delta(t)$ and $\Omega(t)$ have the same boundedness property. Therefore, the system (12.6)-(12.8) is an estimator for θ in the noise free case. The presence of noise does not destabilize the observer. Note that as in [24–26, 29] a complication of the equation (12.6) allows one to ensure observation of $\mathbf{x}(t)$ by $\zeta(t)$, however, as it will be shown later such a nice property is not inherited by an adaptive set observer. This is why the simplified equation (12.6) is considered here. Moreover, since the system (12.7) is a stable time-varying filter, the requirement that the signal $\mathbf{C}^T\Omega^T(t)$ should be PE is related with the same properties of the signal $\mathbf{G}(\mathbf{y}_v(t))$.

12.4.2 Adaptive Set Observer Equations

Usually the signal $\rho(t) \in \Upsilon$ is not measured and not available on-line, thus the observer (12.6)-(12.8) is not realizable. For this case we propose an interval observer based on assumption 2 instead of assumption 1:

$$\dot{\zeta}_o = \mathbf{A}_o\,\zeta_o + \mathbf{B}_o\,\mathbf{u} + \varphi(\mathbf{y}_v) + \mathbf{L}_o(\mathbf{y}_v - \mathbf{C}\,\zeta_o); \tag{12.12}$$

$$\dot{\Omega}_o = [\mathbf{A}_o - \mathbf{L}\,\mathbf{C}]\,\Omega_o - \mathbf{G}(\mathbf{y}_v); \tag{12.13}$$

$$\dot{\widehat{\theta}}_o = -\Gamma_o\,\Omega_o^T \mathbf{C}^T\,(\mathbf{y}_v - \mathbf{C}\,\zeta_o + \mathbf{C}\,\Omega_o\,\widehat{\theta}_o), \Gamma_o = \Gamma_o^T > 0, \tag{12.14}$$

where the index $o \in \{m, M\}$ denotes the upper and lower interval bounds, $\zeta_o \in \mathbb{R}^n$, $\Omega_o \in \mathbb{R}^{n\times q}$ and $\widehat{\theta}_o \in \mathbb{R}^q$ have the same meaning.
In set observer design the monotonicity property of observers equations plays an essential role. As it can be deduced from equations (12.12)-(12.14), the monotonicity of the first two subsystems (12.12), (12.13) is predefined by assumption 2 conditions. Monotonicity of the system (12.14), that defines dynamics of parameters estimator, may not be followed by the same property of the systems (12.12), (12.13). Actually, it is shown below that under some conditions, the dynamics of the system (12.14) can be either cooperative or competitive, impacting the admissible set of θ construction. In the following subsections each case will be analyzed and the new results are summarized in the Theorems 12.1 and 12.2.

12.4.3 Competitive Case

The following theorem establishes stability and monotonicity properties of the observers (12.12)-(12.14) for $o \in \{m, M\}$.

Theorem 12.1. *Let assumption 2 hold, and* $\mathbf{x}(t) \in X$, $\mathbf{u}(t) \in U$, $\mathbf{v}(t) \in V$, $\rho(t) \in \Upsilon$ *and* $\theta \in \Theta$ *for all* $t \geq 0$, *and assume that the signals* $\Omega_o^T(t)\mathbf{C}^T$ *are* (ℓ_o, ϑ_o)*-PE for some* $\ell_o > 0$, $\vartheta_o > 0$, $o \in \{m, M\}$. *Then:*

(i) for all $t \geq 0$ *and* $o \in \{m, M\}$ *the solutions* $\zeta_o(t)$, $\Omega_o(t)$ *and* $\widehat{\theta}_o(t)$ *of the system (12.12)-(12.14) are bounded;*
(ii) if $0 \prec \mathbf{C}$, $\mathbf{v}(t) \equiv 0$ *for all* $t \geq 0$ *and there exists a matrix* $\bar{\Gamma}$ *such that for all* $0 \prec \Gamma_o \prec \bar{\Gamma}$, $o \in \{m, M\}$, *then the following properties hold*
a. if $\Omega_o(0) = 0$, $o \in \{m, M\}$, $\varepsilon_m(0) \geq 0$, $\varepsilon_M(0) \leq 0$, $\widehat{\theta}_M(0) = \theta_m$, $\widehat{\theta}_m(0) = \theta_M$ *and* $\theta_M \leq \mathbf{R}_m^{-1}\mathbf{b}_m$, $\mathbf{R}_M^{-1}\mathbf{b}_M \leq \theta_m$ *where*

$$\mathbf{b}_o = -\lim_{T\to+\infty} T^{-1}\int_0^T \Omega_o^T(t)\mathbf{C}^T\mathbf{C}\varepsilon_o(t)\,dt,$$

$$\mathbf{R}_o = \lim_{T\to+\infty} T^{-1}\int_0^T \Omega_o^T(t)\mathbf{C}^T\mathbf{C}\Omega_o(t)\,dt, o \in \{m, M\},$$

then

$$\widehat{\theta}_M(t) \leq \theta \leq \widehat{\theta}_m(t), t \geq 0.$$

b. if $\Omega_o(0) = 0$, $o \in \{m, M\}$, $\varepsilon_m(0) \leq 0$, $\varepsilon_M(0) \geq 0$, $\widehat{\theta}_m(0) = \theta_m$, $\widehat{\theta}_M(0) = \theta_M$ *and* $\theta_M \leq \mathbf{R}_M^{-1}\mathbf{b}_M$, $\mathbf{R}_m^{-1}\mathbf{b}_m \leq \theta_m$, *then*

$$\widehat{\theta}_m(t) \leq \theta \leq \widehat{\theta}_M(t), t \geq 0.$$

Proof. Define $\varepsilon_o = \mathbf{x} - \zeta_o$, $\tilde{\theta}_o = \theta - \widehat{\theta}_o$ and $\delta_o = \varepsilon_o + \Omega_o\theta$ for $o \in \{m, M\}$, then we obtain

$$\dot{\varepsilon}_o = [\mathbf{A}_o - \mathbf{L}_o\mathbf{C}]\,\varepsilon_o + \mathbf{G}(\mathbf{y}_v)\,\theta + \mathbf{p}_o + \mathbf{d}_v, \tag{12.15}$$

$$\mathbf{p}_o = [\mathbf{A}(\rho(t)) - \mathbf{A}_o]\mathbf{x} + [\mathbf{B}(\rho(t)) - \mathbf{B}_o]\,\mathbf{u},$$

$$\dot{\delta}_o = [\mathbf{A}_o - \mathbf{L}_o\mathbf{C}]\,\delta_o + \mathbf{p}_o + \mathbf{d}_v, \tag{12.16}$$

$$\dot{\tilde{\theta}}_o = \Gamma_o\,\Omega_o^T\,\mathbf{C}^T\,(\mathbf{C}\,\delta_o + \mathbf{v} - \mathbf{C}\,\Omega_o\,\tilde{\theta}_o). \tag{12.17}$$

The new term $\mathbf{p}_o$ appears in (12.15), (12.16) due to the introduction of $\mathbf{A}_o$, $\mathbf{B}_o$ in (12.12)-(12.14). Under assumption 2 for $\mathbf{y} \in Y$, $\mathbf{v} \in V$ all solutions of the system (12.13) are bounded, i.e. there exists $k_{\Omega,o} > 0$ such that $|\Omega_o(t)| \le k_{\Omega,o}$ for all $t \ge 0$. Then $|\mathbf{d}_v| \le [k_\varphi + k_G|\theta| + |\mathbf{L}_o|]|\mathbf{v}|$ and for $\theta \in \Theta$, $\mathbf{v} \in V$ the signal $\mathbf{d}_v$ remains bounded. The signal $\mathbf{p}_o$ is bounded for any $\rho(t) \in \Upsilon$, $\mathbf{x}(t) \in X$, $\mathbf{u}(t) \in U$. Therefore, if assumption 2 is satisfied, the solutions of the system (12.16) are bounded. In addition, if the signal $\mathbf{C}^T\Omega_o^T(t)$ is persistently exciting, then from lemma 1 the system (12.17) solutions remain bounded. Since $\varepsilon_o(t) = \delta_o(t) - \Omega_o(t)\,\theta$ for all $t \ge 0$, the observation error $\varepsilon_o(t)$ is bounded. Therefore, the first part of the theorem is proven, and the solutions of the system (12.15)-(12.17) remain bounded provided that $\mathbf{x}(t) \in X$, $\mathbf{u}(t) \in U$, $\mathbf{v}(t) \in V$, $t \ge 0$. Now, let $\mathbf{v}(t) = 0$ for all $t \ge 0$, that implies $\mathbf{d}_v(t) = 0$, $t \ge 0$. Since $0 \prec \mathbf{G}(\mathbf{y}+\mathbf{v})$ for all $\mathbf{y}(t) \in Y$, $\mathbf{v}(t) \in V$, $t \ge 0$, then monotonicity of the system (12.13) ensures that $\Omega_o(t) \prec 0$ for all $t \ge 0$ and $o \in \{m, M\}$ for $\Omega_o(0) = 0$. In the equation (12.14) the gain matrix $\Gamma_o\Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$, $t \ge 0$ is positive semidefinite and not negative elementwise for both $o \in \{m, M\}$ due to $0 \prec \mathbf{C}$ (the system (12.14) is competitive [7]). The matrix coefficients Γ_o, $o \in \{m, M\}$ define the rate of changes for the variables $\widehat{\theta}_0$. A modification of Γ_o, $o \in \{m, M\}$ does not affect on behavior of the variables $\Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$ and $\Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\varepsilon_o(t)$ (they are defined by the decoupled from (12.14) equations (12.12), (12.13) and their initial conditions). If Γ_o, $o \in \{m, M\}$ are chosen sufficiently small, then the variables $\widehat{\theta}_0(t)$ become "slowly-varying" in the system (12.3), (12.12)-(12.14) and the variables $\Omega_o(t)$ and $\varepsilon_o(t)$ are the "fast" ones. In such conditions, it is possible to apply averaging technique for the equation (12.14) simplification [8, 9]:

$$\dot{\widehat{\theta}}_o(t) = \Gamma_o\,[\mathbf{b}_o - \mathbf{R}_o\,\widehat{\theta}_o(t)]. \tag{12.18}$$

The matrices $\mathbf{R}_o$, $o \in \{m, M\}$ are positive definite due to the PE condition ($\mathbf{R}_o \ge 0.5\,\vartheta_o/\ell_o\,\mathbf{I}_q$ according to lemma A1 from [24]). The system (12.18) is competitive and stable. The solutions of the system (12.18) asymptotically converge to the equilibrium $\widehat{\theta}_o^\infty = \mathbf{R}_o^{-1}\mathbf{b}_o$. If $\theta_M \le \mathbf{R}_m^{-1}\mathbf{b}_m$ and $\mathbf{R}_M^{-1}\mathbf{b}_M \le \theta_m$, then

$$\lim_{t\to+\infty}\widehat{\theta}_m(t) \ge \theta_M, \lim_{t\to+\infty}\widehat{\theta}_M(t) \le \theta_m.$$

For competitive systems this fact implies that for the initial conditions $\tilde{\theta}_m(0) \le 0$, $\tilde{\theta}_M(0) \ge 0$, then $\tilde{\theta}_m(t) \le 0, \tilde{\theta}_M(t) \ge 0$ for all $t \ge 0$. The part (ii).a of the theorem has been proven. The part (ii).b can be proven in the same way. □

Theorem 12.1 establishes the conditions under which the estimation of the set of possible values for θ is guaranteed. These conditions restrict admissible values for initial conditions of the system (12.12)-(12.14) and provide upper bounds for the gains Γ_o, $o \in \{m, M\}$. For the given set X, the conditions $\varepsilon_m(0) \geq 0$, $\varepsilon_M(0) \leq 0$ can be easily realized.

The most restrictive condition of the theorem deals with $\mathbf{R}_o$ and $\mathbf{b}_o$ computation for $o \in \{m, M\}$, they can be computed only asymptotically (afterwards the observer (12.12)-(12.14) runs). However, these quantities can be used to test reliability of the observers. The values $\widehat{\theta}_o^\infty = \mathbf{R}_o^{-1}\mathbf{b}_o$, $o \in \{m, M\}$ can be evaluated and compared on-line with θ_m and θ_M, i.e. the estimates

$$\begin{aligned}\hat{\mathbf{b}}_o(t) &= -t^{-1}\textstyle\int_0^t \Omega_o^T(\tau)\mathbf{C}^T\mathbf{C}\varepsilon_o(\tau)\,d\tau,\\ \widehat{\mathbf{R}}_o(t) &= t^{-1}\textstyle\int_0^t \Omega_o^T(\tau)\mathbf{C}^T\mathbf{C}\Omega_o(\tau)\,d\tau\end{aligned} \tag{12.19}$$

are well defined for all $t \geq \ell_o$, $o \in \{m, M\}$ (by lemma A1 from [24], the matrix $\widehat{\mathbf{R}}_o(t)$ is not singular for $t \geq \ell_o$) and the variable $\theta_o^\infty(t) = \widehat{\mathbf{R}}_o^{-1}(t)\widehat{\mathbf{b}}_o(t)$ can be used for $\widehat{\theta}_o^\infty$ evaluation. Therefore, while the restrictions $\theta_o^\infty(t) \approx \widehat{\theta}_o^\infty$, $o \in \{m, M\}$ required in Theorem 12.1 are satisfied, the observers generate reliable interval estimates for the vector θ.

From another point of view, Theorem 12.1 fixes initial conditions for the systems (12.12)-(12.14), i.e. if the property $\mathbf{x}_m \leq \mathbf{x} \leq \mathbf{x}_M$ holds for all $\mathbf{x} \in X$ for some $\mathbf{x}_m \in \mathbb{R}^n$, $\mathbf{x}_M \in \mathbb{R}^n$, then the conditions of the part (ii).a of Theorem 12.1 are satisfied taken $\xi_m(0) = \mathbf{x}_m$, $\xi_M(0) = \mathbf{x}_M$, $\Omega_o(0) = 0$, $o \in \{m, M\}$, $\widehat{\theta}_m(0) = \theta_m$, $\widehat{\theta}_m(0) = \theta_M$. Therefore, in the system (12.3), (12.12)-(12.14) the unspecified initial conditions are $\mathbf{x}(0) \in X$ only, then $\mathbf{R}_o$ and $\mathbf{b}_o$, $o \in \{m, M\}$ are functions of $\mathbf{x}(0)$ (assuming for simplicity that $\mathbf{v}(t) = 0$). If the system (12.3) is also monotone, then computation of $\mathbf{R}_o$ and $\mathbf{b}_o$, $o \in \{m, M\}$ for the cases $\mathbf{x}(0) \in \{\mathbf{x}_m, \mathbf{x}_M\}$ with $\theta \in \{\theta_m, \theta_M\}$ has to provide worst-case estimates on the values of $\mathbf{R}_o$ and $\mathbf{b}_o$, $o \in \{m, M\}$.

Remark 12.2. The necessity of $\mathbf{R}_o$, $\mathbf{b}_o$, $o \in \{m, M\}$ computation and the idea of the observers (12.12)-(12.14) design can be clarified in other words for the case of assumption 1 ($\mathbf{L}_m = \mathbf{L}_M = \mathbf{L}$), when $\mathbf{x}(t) \geq 0$, $\mathbf{u}(t) \geq 0$ for all $t \geq 0$. In such a situation $\mathbf{p}_m(t) \geq 0$, $\mathbf{p}_M(t) \leq 0$. Define $\mathbf{E}_\Omega = \Omega - \Omega_o$, where Ω is the system (12.7) solution with $\Omega(0) = 0$, then

$$\dot{\mathbf{E}}_\Omega = [\mathbf{A}_o - \mathbf{L}\mathbf{C}]\mathbf{E}_\Omega + [\mathbf{A}(\rho(t)) - \mathbf{A}_o]\Omega.$$

The system (12.7) is stable from assumption 1, cooperative ($\mathbf{A}_m - \mathbf{L}\mathbf{C} \prec \mathbf{A}(\rho(t)) - \mathbf{L}\mathbf{C} \prec \mathbf{A}_M - \mathbf{L}\mathbf{C}$ for all $t \geq 0$ and both $\mathbf{A}_m - \mathbf{L}\mathbf{C}$ and $\mathbf{A}_M - \mathbf{L}\mathbf{C}$ are cooperative from assumption 2) with negative input and zero initial conditions, therefore, $\Omega(t) \prec 0$ for all $t \geq 0$ (indeed, $\dot{\Omega}(0) \leq 0$ and if $\Omega_{i,j}(t)$, $1 \leq i \leq n$, $1 \leq j \leq q$ approaches zero from below, then $\dot{\Omega}_{i,j}(t)$ becomes negative ensuring that $\Omega(t) \prec 0$ for all $t \geq 0$). Thus, $[\mathbf{A}(\rho(t)) - \mathbf{A}_m]\Omega(t) \prec 0$ and $0 \prec [\mathbf{A}(\rho(t)) - \mathbf{A}_M]\Omega(t)$, that under assumption 2 means for $\Omega_o(0) = 0$:

$$\Omega_M(t) \prec \Omega(t) \prec \Omega_m(t) \prec 0, \quad \forall t \geq 0.$$

Cooperativeness of the matrix $\mathbf{A}_o - \mathbf{L}\mathbf{C}$ in the system (12.16) implies that $\delta_m(t) \geq 0$, $\delta_M(t) \leq 0$ for all $t \geq 0$ provided that $\delta_m(0) \geq 0$, $\delta_M(0) \leq 0$ respectively (the conditions $\delta_m(0) \geq 0$, $\delta_M(0) \leq 0$ are satisfied for $\varepsilon_m(0) \geq 0$ and $\varepsilon_M(0) \leq 0$ since $\Omega_o(0) = 0$).

Further, in the equation (12.17) the gain matrix $\Gamma_o \Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$, $t \geq 0$ is positive semidefinite and not negative elementwise for both $o \in \{m, M\}$ (the system (12.17) is competitive [7]), $\Gamma_m \Omega_m^T(t)\mathbf{C}^T\mathbf{C}\,\delta_m(t) \leq 0$ and $\Gamma_M \Omega_M^T(t)\mathbf{C}^T\mathbf{C}\,\delta_M(t) \geq 0$ for all $t \geq 0$. If Γ_o, $o \in \{m, M\}$ are chosen sufficiently small, then the variables $\tilde{\theta}_0(t)$ become "slowly-varying" in the system (12.3), (12.13), (12.15)-(12.17) and the variables $\Omega_o(t)$ and $\delta_o(t)$ are the "fast" ones. Under these conditions, averaging technique gives:

$$\begin{aligned} \dot{\tilde{\theta}}_o(t) &= \Gamma_o[\mathbf{h}_o - \mathbf{R}_o\tilde{\theta}_o(t)], \\ \mathbf{h}_o &= \lim_{T \to +\infty} T^{-1} \textstyle\int_0^T \Omega_o^T(t)\mathbf{C}^T\mathbf{C}\delta_o(t)dt. \end{aligned} \tag{12.20}$$

Note, that $\Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\delta_o(t)$ and $\Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$ are elementwise sign definite functions, therefore, $\mathbf{h}_o$ and $\mathbf{R}_o$ inherit this property, namely

$$\mathbf{h}_m \leq 0 \leq \mathbf{h}_M; \quad \mathbf{R}_o = \mathbf{R}_o^T > 0, 0 \prec \mathbf{R}_o, o \in \{m, M\}.$$

Additionally, since $\Omega_M(t) \prec \Omega_m(t) \prec 0$ for all $t \geq 0$ we have $\mathbf{R}_m \prec \mathbf{R}_M$. Thus, the system (12.20) is competitive and stable. The solutions of the system (12.20) converge asymptotically to the equilibrium $\tilde{\theta}_o^\infty = \mathbf{R}_o^{-1}\mathbf{h}_o$. In addition, if $\mathbf{R}_m^{-1}\mathbf{h}_m \leq 0$ and $\mathbf{R}_M^{-1}\mathbf{h}_M \geq 0$, then

$$\lim_{t \to +\infty} \tilde{\theta}_m(t) \leq 0, \lim_{t \to +\infty} \tilde{\theta}_M(t) \geq 0.$$

For competitive systems this fact implies that for the initial conditions $\tilde{\theta}_m(0) \leq 0$, $\tilde{\theta}_M(0) \geq 0$, $\tilde{\theta}_m(t) \leq 0$, $\tilde{\theta}_M(t) \geq 0$ for all $t \geq 0$, that is exactly the conclusion of part (ii).a of Theorem 12.1 (the part (ii).b can be illustrated by the case $\mathbf{x}(t) \leq 0$, $\mathbf{u}(t) \leq 0$ for all $t \geq 0$).

Unfortunately, all these nice monotonicity properties for $\mathbf{h}_o$ and $\mathbf{R}_o$, $o \in \{m, M\}$ are not enough to ensure $\mathbf{R}_m^{-1}\mathbf{h}_m \leq 0$ and $\mathbf{R}_M^{-1}\mathbf{h}_M \geq 0$ (the inverse matrices $\mathbf{R}_o^{-1}$ are not elementwise sign definite in general case). As a result, the requirement on $\mathbf{R}_o^{-1}\mathbf{b}_o$ on-line checking is introduced in Theorem 12.1.

Remark 12.3. Let us stress that PE property of the signals $\Omega_o^T(t)\mathbf{C}^T$, $o \in \{m, M\}$ can also be checked on-line by computing the integrals

$$\int_t^{t+\ell_o} \Omega_o^T(\tau)\mathbf{C}^T\mathbf{C}\,\Omega_o(\tau)\,d\tau,$$

for $o \in \{m, M\}$ and some $\ell_o > 0$ for all $t \geq 0$. While these integrals result in a nonsingular matrix, the PE property holds. According to lemma A1 in [24], nonsingularity of these integrals are equivalent to the same property of the following integral:

$$t^{-1} \int_0^t \Omega_o^T(\tau) \mathbf{C}^T \mathbf{C} \Omega_o(\tau) d\tau,$$

that coincides with $\widehat{\mathbf{R}}_o(t)$ from (12.19). Thus, by calculating (12.19), it is possible to check on-line PE properties for $\Omega_o^T(t)\mathbf{C}^T$, $o \in \{m, M\}$, simultaneously with verification of the conditions on $\mathbf{R}_o^{-1}\mathbf{b}_o$, $o \in \{m, M\}$.

Remark 12.4. If the functions $\mathbf{C}\Omega_o(t)$ and $\mathbf{C}\varepsilon_o(t)$ are T-periodical, then the limits can be dropped in the definitions of $\mathbf{h}_o$ and $\mathbf{R}_o$, $o \in \{m, M\}$ in Theorem 12.1 formulation. In this case, on-line verification of the conditions for $\mathbf{R}_o^{-1}\mathbf{b}_o$ via (12.19) becomes trivial.

Fulfillment of the conditions $\theta_M \leq \mathbf{R}_m^{-1}\mathbf{b}_m$, $\mathbf{R}_M^{-1}\mathbf{b}_M \leq \theta_m$ or $\theta_M \leq \mathbf{R}_M^{-1}\mathbf{b}_M$, $\mathbf{R}_m^{-1}\mathbf{b}_m \leq \theta_m$ implies that the lower and upper estimates of possible values of $\widehat{\theta}_o$, $o \in \{m, M\}$ lie outside of the admissible values interval $[\theta_m, \theta_M]$ for the vector of unknown parameters θ. However, this fact does not mean that the observer (12.12)-(12.14) can not improve available a priori estimate on the admissible interval $[\theta_m, \theta_M]$. The variables $\widehat{\theta}_o$, $o \in \{m, M\}$ converge to these conservative asymptotic estimates $\mathbf{R}_o^{-1}\mathbf{b}_o$ for sufficiently small values of Γ_o. By closing the gains Γ_o, $o \in \{m, M\}$ to the boundary $\bar{\Gamma}$ it is possible to compute a more accurate estimate on admissible interval values for θ, that we are going to show in the following example.

Example 12.1. Let

$$\mathbf{A}(t) = \begin{bmatrix} -1 + 0.5\sin(t) & 1 & 0 \\ 1.2 & -2 + 0.3\cos(3t) & 1.3 \\ 0 & 1 & -3 + 0.6\cos(2t) \end{bmatrix},$$
$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1\,0\,0 \\ 0\,1\,0 \end{bmatrix},$$
$$\mathbf{G}(t) = \begin{bmatrix} 0 & 1 \\ 1 - 0.2\sin(2t) & 0 \\ 0 & 1 + 0.3\sin(3t) \end{bmatrix}.$$

In this example, we assume that the exact dependence of the matrix $\mathbf{A}$ on time argument is not known and only majorant matrices are available:

$$\mathbf{A}_m = \begin{bmatrix} -1.5 & 1 & 0 \\ 1.2 & -2.3 & 1.3 \\ 0 & 1 & -3.6 \end{bmatrix},$$
$$\mathbf{A}_M = \begin{bmatrix} -0.5 & 1 & 0 \\ 1.2 & -1.7 & 1.3 \\ 0 & 1 & -2.4 \end{bmatrix},$$

while the matrix function $\mathbf{G}(t)$ is measured as it is required in the system (12.1). Assume that

$$\theta(t) = \begin{cases} \theta_1 & \text{if } 0 \leq t \leq t_\theta; \\ \theta_2 & \text{if } t_\theta < t \leq t_f, \end{cases}$$

$$\theta_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \theta_2 = \begin{bmatrix} -1 \\ -2 \end{bmatrix},$$

where $t_f = 600$ is the time of simulation and $t_\theta = 0.5t_f$. Let

$$\mathbf{L}_m = \mathbf{L}_M = \mathbf{L} = \begin{bmatrix} 2\ 0\ 0 \\ 0\ 3\ 1 \end{bmatrix}^T,$$

then assumption 2 holds for

$$\mathbf{A}_m - \mathbf{L}\,\mathbf{C} = \begin{bmatrix} -3.5 & 1 & 0 \\ 1.2 & -5.3 & 1.3 \\ 0 & 0 & -3.6 \end{bmatrix},$$
$$\mathbf{A}_M - \mathbf{L}\,\mathbf{C} = \begin{bmatrix} -2.5 & 1 & 0 \\ 1.2 & -4.7 & 1.3 \\ 0 & 0 & -2.4 \end{bmatrix}$$

and

$$\theta_m = [1\ -4.5]^T, \theta_M = [3.5\ 7]^T$$

for $0 \leq t \leq t_\theta$ and

$$\theta_m = [-2.5\ -9]^T, \theta_M = [0\ 4.5]^T$$

for $t_\theta \leq t \leq t_f$.
Let $\mathbf{x}(0) = [1\ 1\ 1]^T$ and $\Gamma = \Gamma_m = \Gamma_M = 5\mathbf{I}_2$. The results of (12.19) computations and on-line graphical checking the conditions on $\mathbf{R}_o^{-1}\mathbf{b}_o$, Theorem 12.1 are satisfied for $0 \leq t \leq t_\theta$, and conditions of the point (ii).b are satisfied for $t_\theta \leq t \leq t_f$. The variables $\widehat{\theta}$ (the estimate of the ideal observer (12.6)-(12.8)), $\widehat{\theta}_m$ and $\widehat{\theta}_M$ are plotted in Fig. 12.1,c and d for the case without disturbances. The variables $\widehat{\theta}$, $\widehat{\theta}_m$ and $\widehat{\theta}_M$ for the case of a stochastic noise presence with $|\mathbf{v}(t)| \leq 1$ are shown in Fig. 12.2.

Before we continue it is worth to emphasize one feature of the proposed set adaptive observers illustrated by Figs. 12.1 and 12.2. The purpose is not the exact estimation of the values of uncertain parameters, but to evaluate the set or the interval of admissible values for such parameters. Therefore, the lower or upper estimate may have a different sign with respect to the real value of the parameter. The accuracy of this approach consists in the interval length comparing with the ”size” of uncertainty and complexity presented in the estimated system. In the situation when it is possible to design a conventional observer converging to exact values of state $\mathbf{x}$ or parameters $\mathbf{d}$ there is no need in interval observation. However, frequently for complex nonlinear systems with signal and parametric uncertainties the design of conventional exact observers is not possible. In this case the interval observation becomes useful, being the only available solution in practice.

12.4.4 Cooperative Case

Competitiveness of the adaptive observers (12.12)-(12.14) follows by assumption that $0 \prec \mathbf{C}$. Such restriction is natural and corresponds to the situation when some part of the state space vector $\mathbf{x}$ coordinates are available for measurements. Relaxation of this assumption leads to the case when the matrices $\Omega_o^T(t)\,\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$, $o \in \{m,M\}$ may become cooperative.

Theorem 12.2. *Let assumption 2 hold, and* $\mathbf{x}(t) \in X$, $\mathbf{u}(t) \in U$, $\mathbf{v}(t) \in V$, $\rho(t) \in \Upsilon$ *and* $\theta \in \Theta$ *for all* $t \geq 0$, *and assume that the signals* $\Omega_o^T(t)\,\mathbf{C}^T$ *are* (ℓ_o, ϑ_o)*-PE for some* $\ell_o > 0$, $\vartheta_o > 0$, $o \in \{m,M\}$. *Then*

(i) for all $t \in R$ *and* $o \in \{m,M\}$ *the solutions* $\zeta_o(t)$, $\Omega_o(t)$ *and* $\widehat{\theta}_o(t)$ *of the system (12.12)-(12.14) are bounded;*
(ii) let $\mathbf{v}(t) \equiv 0$ *and the matrices* $-\Gamma_o\,\Omega_o^T(t)\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$ *be cooperative for all* $t \geq 0$, $o \in \{m,M\}$,
a. if for all $t \geq 0$ *and* $o \in \{m,M\}$, $O = \{m,M\}\backslash o$,

$$\Gamma_o\,\Omega_o^T(t)\,\mathbf{C}^T\mathbf{C}\,[\varepsilon_o(t) + \Omega_o(t)\,\theta_o] \geq 0,$$

$$\Gamma_o\,\Omega_o^T(t)\,\mathbf{C}^T\mathbf{C}\,\Omega_o(t)\,(\theta_O - \theta_o) \geq 0;$$

$$\Gamma_O\,\Omega_O^T(t)\,\mathbf{C}^T\mathbf{C}\,[\varepsilon_O(t) + \Omega_O(t)\,\theta_O] \leq 0,$$

$$\Gamma_O\,\Omega_O^T(t)\,\mathbf{C}^T\mathbf{C}\,\Omega_O(t)\,(\theta_o - \theta_O) \leq 0,$$

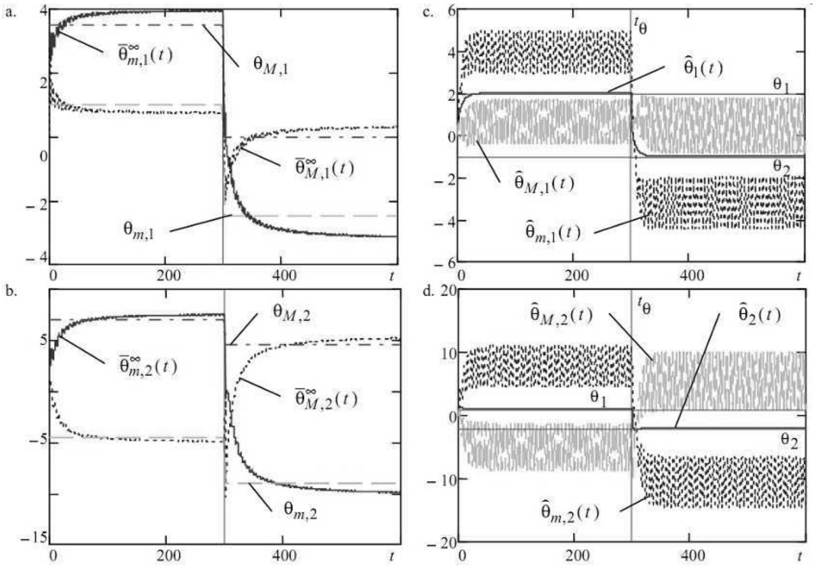


Fig. 12.1: Results of simulation in Example 12.1 (without disturbances)

then $\widehat{\theta}_o(t) \le \theta \le \widehat{\theta}_O(t)$, $t \ge 0$.
b. there exists a matrix $\bar{\Gamma}$ *such that for all* $0 \prec \Gamma_o \prec \bar{\Gamma}$, $o \in \{m, M\}$ *if the signals* $\mathbf{C}\varepsilon_o(t)$ *and* $\mathbf{C}\Omega_o(t)$ *are* T*-periodical for some* $T > 0$, $t \ge 0$ *and for all* $t \ge 0$ *and* $o \in \{m, M\}$, $O = \{m, M\} \backslash o$,

$$\mathbf{b}_o \le \mathbf{R}_o\,\theta_o, \mathbf{R}_o\,(\,\theta_O - \theta_o\,) \ge 0;$$

$$\mathbf{b}_O \ge \mathbf{R}_O\,\theta_O, \mathbf{R}_O\,(\,\theta_o - \theta_O\,) \le 0,$$

then $\widehat{\theta}_o(t) \le \theta \le \widehat{\theta}_O(t)$, $t \ge 0$, *where*

$$\mathbf{b}_o = -T^{-1}\int_0^T \Omega_o^T(\,\tau\,)\mathbf{C}^T\mathbf{C}\,\varepsilon_o(\,\tau\,)\,d\tau,$$

$$\mathbf{R}_o = T^{-1}\int_0^T \Omega_o^T(\,\tau\,)\mathbf{C}^T\mathbf{C}\,\Omega_o(\,\tau\,)\,d\tau.$$

Proof. The part (i) of the theorem can be proven in the same way as in Theorem 12.1. Under conditions of the part (ii).a the system (12.14) is asymptotically stable cooperative with sign definite inputs. Rewriting the system (12.14) equations we obtain for $o \in \{m, M\}$:

$$\begin{gathered}\dot{\tilde{\theta}}_o = -\Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\varepsilon_o - \Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\Omega_o\,\tilde{\theta}_o - \Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\Omega_o\,\theta,\\ \tilde{\theta}_o = \widehat{\theta}_o - \theta.\end{gathered} \tag{12.21}$$

The matrices $-\Gamma_o\,\Omega_o^T(\,t\,)\,\mathbf{C}^T\mathbf{C}\,\Omega_o(\,t\,)$, $o \in \{m, M\}$ are cooperative and stable (persistency of excitation ensures the last property). If the signals

$$-\Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\delta_o = -\Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\varepsilon_o - \Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\Omega_o\,\theta,$$

for $o \in \{m, M\}$ are sign definite, then by applying monotonicity, it is possible to substantiate desired relations between $\widehat{\theta}_m(\,t\,)$, $\widehat{\theta}_M(\,t\,)$ and θ. Being able to estimate sign of the signal $\delta_o(\,t\,)$, $o \in \{m, M\}$, unfortunately, the signal $-\Gamma_o\,\Omega_o^T\mathbf{C}^T\mathbf{C}\,\delta_o$ is not available for measurements and it is required to evaluate its sign based on given

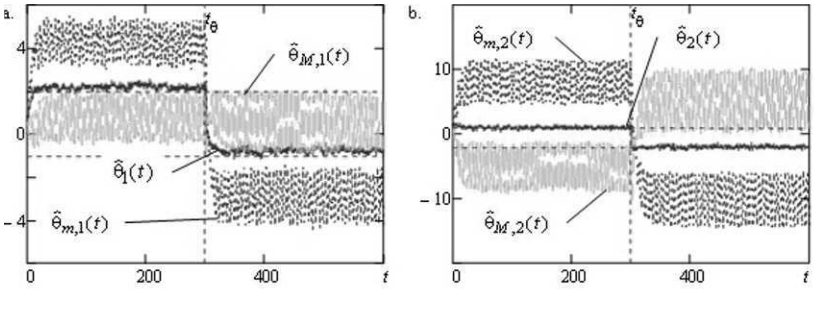


Fig. 12.2: Results of simulation in Example 12.1 (with disturbances)

measurable information. Note that

$$-\Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \delta_o = -\Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \varepsilon_o - \Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \Omega_o\, \theta_o - \Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \Omega_o\, (\,\theta - \theta_o\,),$$

the sign of the signals

$$-\Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \varepsilon_o - \Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \Omega_o\, \theta_o, o \in \{\,m, M\,\}$$

can be verified on-line, while the sign of the last term for all $\theta_m \leq \theta \leq \theta_M$ lies between zero and the sign of

$$\Gamma_o \Omega_o^T \mathbf{C}^T \mathbf{C}\, \Omega_o\, (\,\theta_O - \theta_o\,), o \in \{\,m, M\,\}, O = \{\,m, M\,\} \backslash o,$$

where the symbol is used for the set complement. Therefore, the set of implications hold:

$$-\Gamma_o \Omega_o^T(\,t\,)\, \mathbf{C}^T \mathbf{C}\, \varepsilon_o(\,t\,) - \Gamma_o \Omega_o^T(\,t\,)\, \mathbf{C}^T \mathbf{C}\, \Omega_o(\,t\,)\, \theta_o \leq 0,$$
$$-\Gamma_o \Omega_o^T(\,t\,)\, \mathbf{C}^T \mathbf{C}\, \Omega_o(\,t\,)\, (\,\theta_O - \theta_o\,) \leq 0, t \geq 0 \Rightarrow \widehat{\theta}_o(\,t\,) \leq \theta;$$

$$-\Gamma_o \Omega_o^T(\,t\,)\, \mathbf{C}^T \mathbf{C}\, \varepsilon_o(\,t\,) - \Gamma_o \Omega_o^T(\,t\,)\, \mathbf{C}^T \mathbf{C}\, \Omega_o(\,t\,)\, \theta_o \geq 0,$$
$$-\Gamma_o \Omega_o^T(\,t\,)\, \mathbf{C}^T \mathbf{C}\, \Omega_o(\,t\,)\, (\,\theta_O - \theta_o\,) \geq 0, t \geq 0 \Rightarrow \widehat{\theta}_o(\,t\,) \geq \theta,$$

that implies the theorem claim (ii).a.
To prove part (ii).b, assume that the norm of the matrices Γ_o, $o \in \{\,m, M\,\}$ are chosen small enough to ensure that the variables $\widehat{\theta}_o(\,t\,)$ are slowly-varying in the system (12.12)-(12.14). Applying averaging technique for the equation (12.21) with T-periodical right hand side [8,9] we obtain:

$$\dot{\tilde{\theta}}_o = \mathbf{b}_o - \mathbf{R}_o\, \tilde{\theta}_o - \mathbf{R}_o\, \theta, o \in \{\,m, M\,\},$$

where the matrices $\mathbf{R}_o$, $o \in \{\,m, M\,\}$ are cooperative and Hurwitz by the same arguments. Again

$$\mathbf{b}_o - \mathbf{R}_o\, \theta = \mathbf{b}_o - \mathbf{R}_o\, \theta_o - \mathbf{R}_o\, (\,\theta - \theta_o\,)$$

and the sign of $\mathbf{b}_o - \mathbf{R}_o\, \theta_o$ can be verified during or before the observers operation and $\mathbf{R}_o\, (\,\theta - \theta_o\,) \in [\,0, \mathbf{R}_o\, (\,\theta_O - \theta_o\,)\,]$ for all $\theta_m \leq \theta \leq \theta_M$ and $o \in \{\,m, M\,\}$, $O = \{\,m, M\,\} \backslash o$. □

The cooperative case is more sophisticated and it requires an on-line verification of a bigger number of conditions. To check constraints imposed on $\mathbf{b}_o$, $\mathbf{R}_o$, $o \in \{\,m, M\,\}$ for the system (12.3) solutions being T-periodical asymptotically, the following variables can be computed for $t > T$:

$$\widehat{\mathbf{b}}_o(t) = -T^{-1} \int_{t-T}^{t} \Omega_o^T(\,\tau\,)\, \mathbf{C}^T \mathbf{C}\, \varepsilon_o(\,\tau\,)\, d\tau,$$

$$\widehat{\mathbf{R}}_o(\,t\,) = T^{-1} \int_{t-T}^{t} \Omega_o^T(\,\tau\,)\, \mathbf{C}^T \mathbf{C}\, \Omega_o(\,\tau\,)\, d\tau.$$

Example 12.2. Let

$$\mathbf{A}(t) = \begin{bmatrix} -1+0.1\sin(3t) & 1 & 0.4+0.2\sin(3t) \\ 0 & -1+0.3\cos(t) & 1 \\ 0.5+0.1\cos(2t) & 1 & -2+0.2\cos(2t) \end{bmatrix},$$
$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1\ 0\ -1 \\ 1\ 1\ 0 \end{bmatrix},$$
$$\mathbf{G}(t) = \begin{bmatrix} 1 & 0 \\ 0.3+0.3\sin(2t) & 0 \\ 0 & 0.3+0.2\sin(3t) \end{bmatrix}.$$

Again, in this example we assume that the exact dependence of the matrix **A** on time argument is not known and only majorant matrices are available:

$$\mathbf{A}_m = \begin{bmatrix} -0.9 & 1 & .6 \\ 0 & -0.7 & 1 \\ 0.6 & 1 & -1.8 \end{bmatrix},$$
$$\mathbf{A}_M = \begin{bmatrix} -1.1 & 1 & 0.2 \\ 0 & -1.3 & 1 \\ 0.4 & 1 & -2.2 \end{bmatrix},$$

while the matrix function $\mathbf{G}(t)$ is measured. Assume that

$$\theta(t) = \begin{cases} \theta_1 \text{ if } 0 \le t \le t_\theta\,; \\ \theta_2 \text{ if } t_\theta < t \le t_f\,, \end{cases}$$
$$\theta_1 = \begin{bmatrix} -.5 \\ -1 \end{bmatrix}, \theta_2 = \begin{bmatrix} 0 \\ -2 \end{bmatrix},$$

where $t_f = 600$ is the time of simulation and $t_\theta = 0.5t_f$. Let

$$\mathbf{L}_m = \begin{bmatrix} 0\ -1\ 0 \\ 0.5\ 1\ -1 \end{bmatrix}^T, \mathbf{L}_M = \begin{bmatrix} 0\ -1\ 0 \\ 1\ 1\ 0.6 \end{bmatrix}^T,$$

then assumption 2 holds for $\theta_m = [-1\ \ -2.5]^T$, $\theta_M = [0.5\ 0]^T$ and

$$\mathbf{A}_m - \mathbf{L}_m\mathbf{C} = \begin{bmatrix} -1.6 & 0.5 & 0.2 \\ 0 & -2.3 & 0 \\ 1.4 & 2 & -2.2 \end{bmatrix},$$
$$\mathbf{A}_M - \mathbf{L}_M\mathbf{C} = \begin{bmatrix} -1.9 & 0 & 0.6 \\ 0 & -1.7 & 0 \\ 0 & 0.4 & -1.8 \end{bmatrix}.$$

Let $\mathbf{x}(0) = [0\,0\,0]^T$ and $\Gamma = \Gamma_m = \Gamma_M = diag([40\,180]^T)$. From the system equations we conclude that the solutions become asymptotically 2π-periodical functions of time. Numerical calculations show that $\mathbf{G}(t)$ is persistently excited with $\ell = 2\pi$, therefore the signals $\Omega_o^T(t)\,\mathbf{C}^T$, $o \in \{m, M\}$ possess the same property. Numerical calculation of the matrices $-\Gamma_o\,\Omega_o^T(t)\,\mathbf{C}^T\mathbf{C}\,\Omega_o(t)$, $\widehat{\mathbf{b}}_o(t)$, $\widehat{\mathbf{R}}_o(t)$ for both $o \in \{m, M\}$ shows that the conditions

$$\widehat{\mathbf{b}}_m(t) \le \widehat{\mathbf{R}}_m(t)\,\theta_m, \widehat{\mathbf{R}}_m(t)\,(\theta_M - \theta_m) \ge 0;$$

$$\widehat{\mathbf{b}}_M(t) \ge \widehat{\mathbf{R}}_M(t)\,\theta_M, \widehat{\mathbf{R}}_M(t)\,(\theta_m - \theta_M) \le 0$$

are satisfied for all $t \ge 25$ (the first 25 seconds is the interval of the observer convergence from the chosen zero initial conditions). Therefore, all conditions of Theorem 12.2, part (ii).b hold and it should be $\widehat{\theta}_m(t) \le \theta \le \widehat{\theta}_M(t)$, $t \ge 25$, that is confirmed by results of the system simulation presented in Fig. 12.3. The variables $\widehat{\theta}_m$ and $\widehat{\theta}_M$ for the case of a stochastic noise presence with $|\mathbf{v}(t)| \le 0.5$ are plotted in Fig. 12.4.

Remark 12.5. It is important to note that the conditions of assumption 2 used in Theorems 12.1,12.2 to substantiate properties of the adaptive set observers are less restrictive than the corresponding conditions of assumption 1 applicable to the conventional adaptive observers (it is hard to compute the matrices **L** and **P** from as-

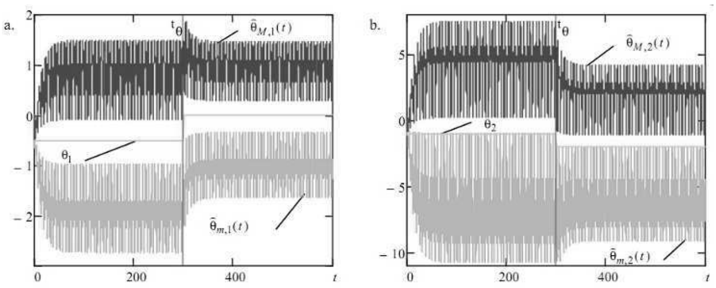


Fig. 12.3: Results of simulation in Example 12.2 (without disturbances)

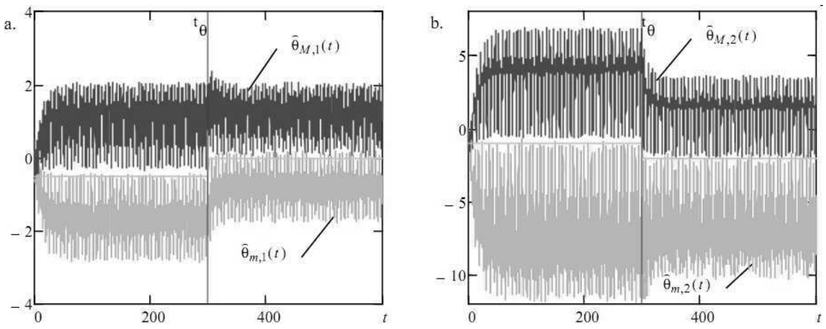


Fig. 12.4: Results of simulation in Example 12.2 (with disturbances)

sumption 1 in general case). This fact justifies that the set observers can be applied in case where conventional observers can not be realized due to lack of information about the system or plant models complexity.

12.5 Set State Observer

Consider the following observers for $o, O_o \in \{m, M\}$:

$$\dot{\xi}_o = \mathbf{A}_o \xi_o + \mathbf{B}_o \mathbf{u} + \varphi(\mathbf{y}_v) + \mathbf{G}(\mathbf{y}_v)\widehat{\theta}_{O_o} + \mathbf{L}_o(\mathbf{y}_v - \mathbf{C}\xi_o), \tag{12.22}$$

where $\widehat{\theta}_{O_o}$, $O_o \in \{m, M\}$ are generated by (12.14) and $\xi_o \in R^n$, $o \in \{m, M\}$ are the state estimates. The equation (12.22) partly repeats (12.12), however, the state ζ_o, $o \in \{m, M\}$ of the system (12.12) can not be used for the state $\mathbf{x}$ interval estimation since one of the inequalities $\widehat{\theta}_m < \widehat{\theta}_M$ or $\widehat{\theta}_M < \widehat{\theta}_m$ holds depending on the auxiliary conditions formulated in Theorems 12.1 and 12.2. This is why an additional index O_o is introduced in (12.22). Under conditions of Theorems 12.1,12.2 the state interval observation via (12.22) follows by standard arguments [7].

Theorem 12.3. *Let assumption 2 hold, and* $\mathbf{x}(t) \in X$, $\mathbf{u}(t) \in U$, $\mathbf{v}(t) \in V$, $\rho(t) \in \Upsilon$ *and* $\theta \in \Theta$ *for all* $t \geq 0$, *and assume that the signals* $\Omega_o^T(t)\mathbf{C}^T$ *are* (ℓ_o, ϑ_o)*-PE for some* $\ell_o > 0$, $\vartheta_o > 0$, $o \in \{m, M\}$. *Then*

(i) for all $t \geq 0$ *and* $o \in \{m, M\}$ *the solutions* $\xi_o(t)$, $\zeta_o(t)$, $\Omega_o(t)$ *and* $\widehat{\theta}_o(t)$ *of the system (12.12)-(12.14), (12.22) are bounded;*
(ii) let $\mathbf{v}(t) \equiv 0$, $\mathbf{x}(t) \geq 0$, $\mathbf{u}(t) \geq 0$ *for all* $t \geq 0$ *and Theorem 12.1, part (ii) or Theorem 12.2, part (ii) conditions are verified indicating that* $\widehat{\theta}_o(t) \leq \theta \leq \widehat{\theta}_O(t)$, $o, O \in \{m, M\}$, $t \geq 0$, *then also* $\xi_m(t) \leq \mathbf{x}(t) \leq \xi_M(t)$ *for all* $t \geq 0$ *provided that* $\xi_m(0) \leq \mathbf{x}(0) \leq \xi_M(0)$ *and* $O_m = o$, $O_M = O$ *in (12.22);*
(iii) let $\mathbf{v}(t) \equiv 0$, $\mathbf{x}(t) \leq 0$, $\mathbf{u}(t) \leq 0$ *for all* $t \geq 0$ *and Theorem 12.1, part (ii) or Theorem 12.2, part (ii) conditions are verified indicating that* $\widehat{\theta}_o(t) \leq \theta \leq \widehat{\theta}_O(t)$, $o, O \in \{m, M\}$, $t \geq 0$, *then also* $\xi_M(t) \leq \mathbf{x}(t) \leq \xi_m(t)$ *for all* $t \geq 0$ *provided that* $\xi_M(0) \leq \mathbf{x}(0) \leq \xi_m(0)$ *and* $O_m = O$, $O_M = o$ *in (12.22).*

Proof. Consider the estimation errors $\mathbf{e}_o = \mathbf{x} - \xi_o$, $o, O_o \in \{m, M\}$,

$$\dot{\mathbf{e}}_o = [\mathbf{A}_o - \mathbf{L}_o\mathbf{C}]\mathbf{e}_o + \mathbf{G}(\mathbf{y}_v)[\theta - \widehat{\theta}_{O_o}] + \mathbf{d}_v + \mathbf{p}_o, \tag{12.23}$$

$$\mathbf{p}_o = [\mathbf{A}(\rho(t)) - \mathbf{A}_o]\mathbf{x} + [\mathbf{B}(\rho(t)) - \mathbf{B}_o]\mathbf{u},$$

$$\mathbf{d}_v = \varphi(\mathbf{y}) - \varphi(\mathbf{y}_v) + [\mathbf{G}(\mathbf{y}) - \mathbf{G}(\mathbf{y}_v)]\theta - \mathbf{L}\mathbf{v}.$$

Since all conditions of Theorem 12.1, part (i) or Theorem 12.2, part (i) are satisfied, then the solutions $\zeta_o(t)$, $\Omega_o(t)$ and $\widehat{\theta}_o(t)$ are bounded for both $o \in \{m, M\}$. While $\mathbf{x}(t) \in X$, $\mathbf{u}(t) \in U$, $\mathbf{v}(t) \in V$, $\rho(t) \in \Upsilon$ and $\theta \in \Theta$ the signals $\mathbf{p}_o(t)$, $o \in \{m, M\}$ and $\mathbf{d}_v(t)$ stay bounded, and under assumption 2, (12.23) is an asymptotically stable cooperative linear system with bounded input $\mathbf{G}(\mathbf{y}_v)[\theta - \widehat{\theta}_{O_o}] + \mathbf{d}_v + \mathbf{p}_o$, that implies boundedness of the variables $\xi_o(t)$, $o \in \{m, M\}$. The part (i) has been proven. To substantiate the part (ii) note that in this case $\mathbf{p}_m(t) \geq 0$, $\mathbf{p}_M(t) \leq 0$, $\mathbf{d}_v(t) = 0$ for $t \geq 0$. Then the system (12.23) with $o = m$ is cooperative with positive input $\mathbf{G}(\mathbf{y})[\theta - \widehat{\theta}_o] + \mathbf{p}_m$, by standard arguments in this case, if $\mathbf{e}_m(0) \geq 0$, then the property $\mathbf{e}_m(t) \geq 0$ is preserved for all $t \geq 0$. For $o = M$ the system (12.23) is cooperative with negative valued input $\mathbf{G}(\mathbf{y})[\theta - \widehat{\theta}_O] + \mathbf{p}_M$, that for $\mathbf{e}_M(0) \leq 0$ implies $\mathbf{e}_M(t) \leq 0, t \geq 0$. In the case of part (iii), $\mathbf{p}_M(t) \geq 0$, $\mathbf{p}_m(t) \leq 0$, $\mathbf{d}_v(t) = 0$ for all $t \geq 0$. Then the input $\mathbf{G}(\mathbf{y})[\theta - \widehat{\theta}_O] + \mathbf{p}_m$ is negative and the input $\mathbf{G}(\mathbf{y})[\theta - \widehat{\theta}_o] + \mathbf{p}_M$ is positive, that implies the theorem claim. □

For easy reference, the computational procedure is summarized as follows:

- Take the given sets X, U, V, Y, Θ, Υ and compute the bounds $\mathbf{x}_m$, $\mathbf{x}_M$, θ_m and θ_M.
- Transform the system (12.1) to the LPV form (12.3).
- Find the matrices $\mathbf{L}_o$, $o \in \{m, M\}$ and verify Assumption 2.
- Build the set adaptive observer (12.12)-(12.14). Calculate (12.19) and check the PE condition. Distinguish competitive or cooperative cases:
  - Competitive case ($0 \prec \mathbf{C}$). Verify the properties of either $\overline{\theta}_o^\infty$ or $\widehat{\theta}_o^\infty$, $o \in \{m, M\}$ in accordance with the part (ii) of Theorem 12.1.
  - Cooperative case (the matrix $-\Gamma_o \Omega_o^T(t) \mathbf{C}^T \mathbf{C} \Omega_o(t)$, $t \geq 0$ is cooperative). Check the inequalities of the part (ii) of Theorem 12.2.
- Augment the set state observer (12.22) and check the conditions of the parts (ii) or (iii) of Theorem 12.3.

Example 12.3. Consider a double mass model for a vibration crusher [31], the masses correspond to two platforms connected by springs and excited by rotating motors. We assume that movements of platforms are possible in vertical plane only. This system is described by:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\beta_1/m(t)x_2 - c/m(t)(x_1 - x_3) - c_0/m(t)x_1 + \theta_1 u_1(t) + \theta_2 u_2(t) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = -\beta_2/M(t)x_4 + c/M(t)(x_1 - x_3) - c_1/M(t)x_3 + \theta_3 u_1(t) + \theta_4 u_2(t) \\ y_1 = x_1 + v_1 \\ y_2 = x_3 + v_2 \end{cases}; \tag{12.24}$$

where $x_1 \in \mathbb{R}$, $x_3 \in \mathbb{R}$ are displacements of the platforms from their steady state positions, $\dot{x}_1 \in \mathbb{R}$, $\dot{x}_3 \in \mathbb{R}$ are velocities of the platforms; $y_1 \in \mathbb{R}$, $y_2 \in \mathbb{R}$ are noisy measurements; u_1, u_2 are exciting forces formed by the rotating motors located on the platforms; β_1, β_2 are small known friction coefficients; values of spring stickiness c_1, c_0 are known, the value c of coupling stickiness is unknown; $\theta \in \mathbb{R}^4$ is the vector of unknown control gains. Values of masses m and M are assumed unknown and time-varying. Bounds are given for all unknown parameters and the state $\mathbf{x}$: $c_m \le c \le c_M$, $m_m \le m(t) \le m_M$, $m_m \le M(t) \le m_M$, $\theta_m \le \theta \le \theta_M$, $\mathbf{x}_m \le \mathbf{x} \le \mathbf{x}_M$. The controls are the positive half-period square pulses with amplitude 1 and periods 5 and 6 respectively. Take

$$\mathbf{A}_m = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -(\beta_1 + c_M)m_m^{-1} & -c_0 m_m^{-1} & c_m m_M^{-1} & 0 \\ 0 & 0 & 0 & 1 \\ c_m m_M^{-1} & 0 & -(\beta_2 + c_M)m_m^{-1} & -c_0 m_m^{-1} \end{bmatrix},$$

$$\mathbf{A}_M = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -(\beta_1 + c_m)m_M^{-1} & -c_0 m_M^{-1} & c_M m_m^{-1} & 0 \\ 0 & 0 & 0 & 1 \\ c_M m m & 0 & -(\beta_2 + c_m)m_M^{-1} & -c_0 m_M^{-1} \end{bmatrix},$$

$$\mathbf{G}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ u_1(t) & u_2(t) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & u_1(t) & u_2(t) \end{bmatrix},$$

$$\mathbf{L}_m \begin{bmatrix} 1 & 0 \\ -(\beta_1 + c_M)m_m^{-1} & 0 \\ 0 & 1 \\ 0 & -(\beta_2 + c_M)m_m^{-1} \end{bmatrix},$$

$$\mathbf{L}_M = \begin{bmatrix} 1 & 0 \\ -(\beta_1 + c_m)m_M^{-1} & 0 \\ 0 & 1 \\ 0 & -(\beta_2 + c_m)m_M^{-1} \end{bmatrix},$$

$\mathbf{B} = 0$, $\phi(y) = 0$, then the matrices $\mathbf{A}_o - \mathbf{L}_o\mathbf{C}$, $o \in \{m, M\}$ are cooperative and asymptotically stable (assumption 2 is satisfied). For the parameters

$$\begin{aligned} & m_m = 0.25, m_M = 0.33; c_m = 0.08, c_M = 0.12, c = 0.1; \\ & \theta_m = [0.5\ 0\ 0\ 0.5]^T, \theta_M = [2\ 1\ 1\ 2]^T, \theta = [1\ 0.5\ 0.5\ 1.3]^T, \\ & M(t) = m_M^{-1} + m_m^{-1} - m(t), \\ & m(t) = 0.5(m_M^{-1} - m_m^{-1})(1 + 0.1(t - 0.5t_k)/[1 + 0.1|t - 0.5t_k|]) + m_m^{-1} + 0.05\sin(3t), \end{aligned}$$

where $t_k = 100$ is the simulation time interval. The results of the parameter θ estimation are shown in 12.5 and the estimates provided by the state observer are plotted in 12.6.

Remark 12.6. The requirement imposed in Theorems 12.1–12.3 on initial conditions $\xi_o(0)$, $\zeta_M(0)$, $\Omega_o(0)$, $\widehat{\theta}_o(0)$, $o \in \{m, M\}$ are not restrictive and can be skipped, that may result in additional transients in the intervals evaluation (for linear stable systems the asymptotic behavior is defined by properties of external inputs).

Remark 12.7. An advantage of the designed solution is that exponential complexity often met with set-membership parameter estimation is avoided. In [16,22,23], the problem is formulated as a Constraint Satisfaction Problem (CSP) involving an ordinary differential equation. The CSP is solved in a rigorous way using branch and bound algorithms. The main particularity of these techniques is that the parameter domain is systematically partitioned at each iteration that makes the complexity exponential with respect to the dimension of the parameter vector. It has been proven that the number of iterations is given by:

$$N = (W([\Theta])/\varepsilon + 1)^q,$$

where $W([\Theta])$ is the width of the domain of the parameter vector θ (a measure of the set Θ); ε is a tolerance fixed by the user in order to have a result in a finite time, and q is the dimension of the parameter vector. In addition, it is important to note that each iteration should be solved for all the instants of time t_j, where $j \geq 0$ lies in the range of the interval of simulation. This process is known to be time-consuming. This limitation is avoided in our work and the complexity of the proposed observer is $2(2n + n \times q + q)$, that is similar to the Kalman filter. This achievement makes

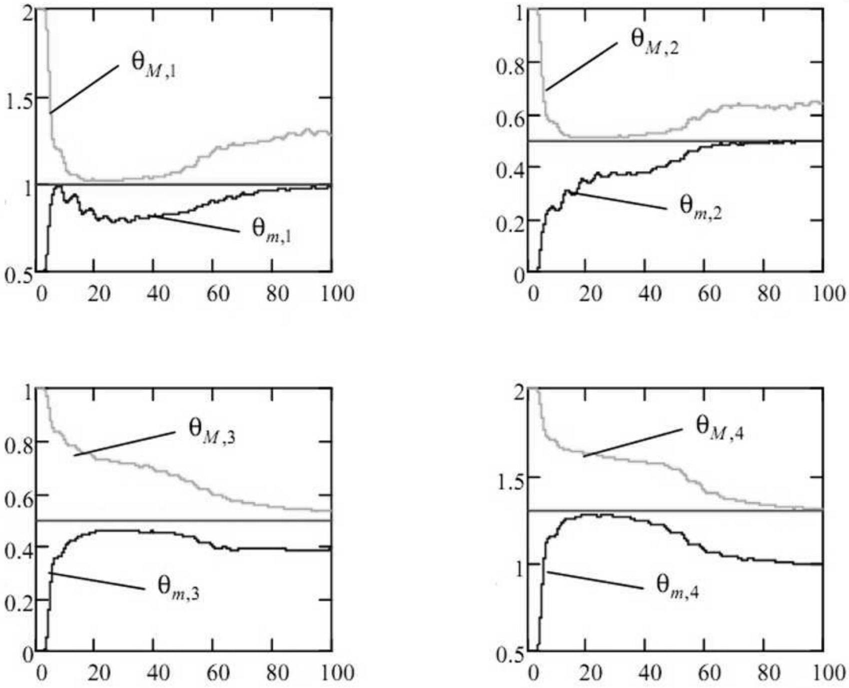


Fig. 12.5: Results of parameter estimation for the model (12.24)

reasonable application of the proposed observer to high dimensional uncertain nonlinear systems.

12.6 Conclusion

The basic problem studied in this work is adaptive observer design for joint parameter and state estimation of nonlinear continuous time systems. Based on a LPV approximation, the problem of set observer design for the nonlinear system is reformulated in terms of adaptive observer design for LPV ones. The exponential complexity, often met, for set-membership parameter estimation in nonlinear continuous-time systems is avoided. The complexity of the proposed observer is similar to the Kalman filter and the dimension of the set adaptive observer equations increases proportionally to the parameter θ and to the state $\mathbf{x}$ dimensions (the full adaptive set observer dimension is $2\,(2n+n\times q+q)$). This setting makes possible the application of the proposed observer for high dimension uncertain systems. It is shown that under standard cooperativity assumption imposed on the observer equations, the adaptation loop may be cooperative or competitive depending on additional circumstances. Both competitive and cooperative cases are analyzed and

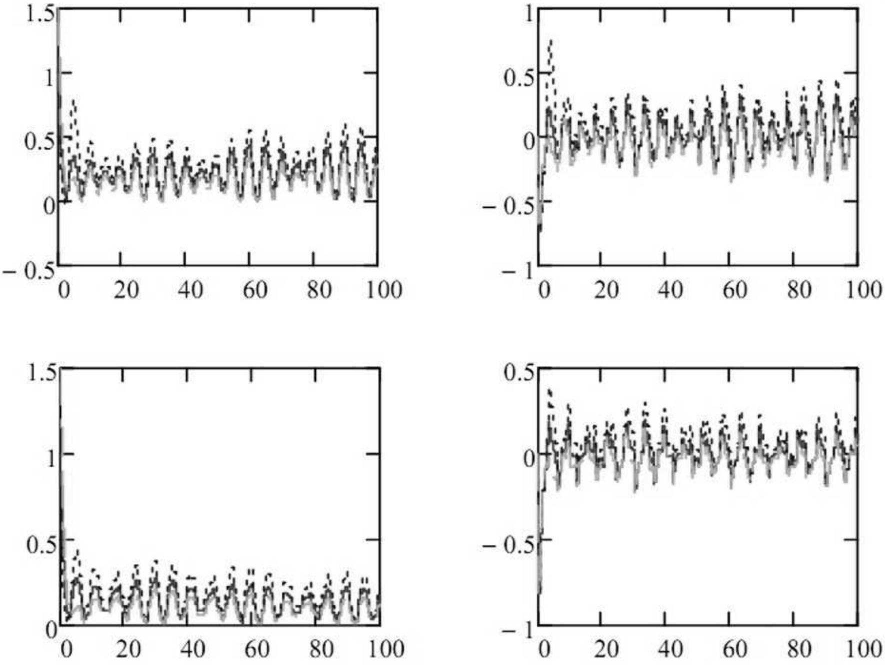


Fig. 12.6: Results of state estimation for the model (12.24)

applicability conditions for the adaptive observers are proposed. Moreover, the proposed applicability conditions of the adaptive set observers (presented in Assumption 2) are less restrictive than those corresponding to the conventional adaptive observers (formulated in Assumption 1). Thus, the adaptive set observers can be applied in the cases when the solution of the parameter dependent Lyapunov equation from Assumption 1 is not feasible.

The results of the developed techniques suggest that in the presence of small uncertainties (small deviations of the parameters and the state from their nominal/majorant values) the introduction of adaptive technology may not provide significant improvement in the state estimation. However, if the set of admissible values for the model parameters is largely deviated or under noisy conditions, then the adaptive set observers proposed here could be superior to the already existing solutions.

References

1. Besançon, G.: Nonlinear observers and applications. Lecture Notes in Control and Inforamtion Science. Springer Verlag, Berlin (2007)
2. Nijmeijer H., Fossen, T.I.: New Directions in Nonlinear Observer Design. Springer-Verlag, London (1999)
3. Lee, L.H.: Identification and Robust Control of Linear Parameter-Varying Systems. PhD-thesis, University of California at Berkeley, Berkeley, California (1997)
4. Tan, W.: Applications of Linear Parameter-Varying Control Theory. PhD-thesis, Dept. of Mechanical Engineering, University of California at Berkeley (1997)
5. Hansen, R.E.: Global optimization using interval analysis. CRC, 2nd edition (2004)
6. Moore, R.E.: Interval analysis. Prentice-Hall, Englewood Cliffs, NJ (1966)
7. Smith, H.L.: Monotone Dynamical Systems: An Introduction to the Theory of Competitive and Cooperative Systems. AMS, Providence (1995)
8. Bogoliubov, N.N., Mitropolskii, Yu.A.: Asymptotic methods in the theory of nonlinear oscillations. Gordon and Breach, New York (1961)
9. Sanders, J., Verhulst, F., Murdock, J.: Averaging Methods in Nonlinear Dynamical Systems. Springer, New York (2007)
10. Kletting, M., Verified Methods for State and Parameter Estimators for Nonlinear Uncertain Systems with Applications in Engineering. PhD-thesis, Institute of Measurement, Control, and Microtechnology, University of Ulm, Germany (2009)
11. Bokor, J., Balas, G.: Detection Filter Design for LPV Systems - a Geometric Approach. Automatica **40**, 511-518 (2004)
12. Marcos, A., Balas, G.: Development of linear-parameter-varying models for aircraft. J. Guidance, Control, Dynamics **27(2)**, 218-228 (2004)
13. Shamma, J., Cloutier, J.: Gain-scheduled missile autopilot design using linear parameter-varying transformations. J. Guidance, Control, Dynamics **16(2)**, 256-261 (1993)
14. Raïssi, T., Videau, G., Zolghadri, A.: Interval observers design for consistency checks of nonlinear continuous-time systems. Automatica **46(3)**, 518-527 (2010)
15. Jaulin, L.: Nonlinear bounded-error state estimation of continuous time systems. Automatica **38(6)**, 1079-1082 (2002)
16. Raïssi, T., Ramdani, N., Candau, Y.: Set membership state and parameter estimation for systems described by nonlinear differential equations. Automatica **40(10)**, 1771-1777 (2004)
17. Kieffer, K. Walter, E.: Guaranteed nonlinear state estimator for cooperative systems. Numerical Algorithms **37**, 187-198 (2004)

18. Müller, M.: Über das fundamental theorem in der theorie der gewöhnlichen differentialgleichungen. Math. Z **26**, 619-645 (1920)
19. Bernard, O., Gouzé, J.L.: Closed loop observers bundle for uncertain biotechnological models. J. Process Control **14**, 765-774 (2004)
20. Gouzé, J.L., Rapaport, A., Hadj-Sadok, M.Z.: Interval observers for uncertain biological systems. Ecological Modeling **133**, 46-56 (2000)
21. Moisan, M., Bernard, O., Gouzé, J.L.: Near optimal interval observers bundle for uncertain bioreactors. Automatica **45(1)**, 291-295 (2009)
22. Jaulin, L., Walter, E.: Set inversion via interval analysis for nonlinear bounded-error estimation. Automatica **29(4)**, 1053-1064 (1993)
23. Johnson, T., Tucker, W.: Rigorous parameter reconstruction for differential equations with noisy data. Automatica **44(9)**, 2422-2426 (2008)
24. Efimov, D.: Dynamical adaptive synchronization, Int. J. Adaptive Control and Signal Processing **20(9)**, 491-507 (2006)
25. Xu, A., Zhang, Q.: Residual Generation for Fault Diagnosis in Linear Time-Varying Systems. IEEE Trans. Autom. Control **49(5)**, 767-772 (2004)
26. Zhang, Q.: Adaptive observer for multiple-input-multiple-output (MIMO) linear time varying systems. IEEE Trans. Autom. Control **(47(3)**, 525-529 (2002)
27. Sontag, E.D., Wang, Y.: Notions of input to output stability. Systems and Control Letters **38**, 235-248 (1999)
28. Efimov, D., Raïssi, T., Zolghadri, A.: Adaptive set observers design for nonlinear continuous-time systems: Application to fault detection and diagnosis. IEEE Trans. Autom. Control, revised.
29. Fradkov, A.L., Nikiforov, V.O., Andrievsky, B.R.: Adaptive observers for nonlinear nonpassifiable systems with application to signal transmission. Proc. 41th IEEE Conf. Decision and Control, 4706-4711, Las Vegas (2002)
30. Meslem, N., Ramdani, N., Candau, Y.: Interval Observers for Uncertain Nonlinear Systems. Application to bioreactors. Proc. 17th IFAC World Congress, 9667-9672, Seoul, (2008)
31. Efimov D.V., Fradkov A.L. Hybrid adaptive resonance control of vibration machines: the double mass case. Proc. 3rd IFAC Workshop Periodic Control Systems (PSYCO'07), Saint-Petersburg, (2007)

Chapter 13
Nonlinear Adaptive Control of a Bioprocess Model with Unknown Kinetics

Neli S. Dimitrova (✉) and Mikhail I. Krastanov

Abstract In this paper we consider a nonlinear model of an anaerobic wastewater treatment process, in which biodegradable organic is decomposed to produce methane. The model, described by a four-dimensional dynamic system, is known to be practically validated and reliable. We propose a feedback control law for asymptotic stabilization of the closed-loop system towards a fixed operating point. Moreover, a model-based numerical extremum seeking algorithm is applied to stabilize the control system towards an equilibrium point with maximal methane flow rate. The robustness of the feedback control is demonstrated by assuming uncertainties in the growth rate functions. Computer simulations are reported to illustrate the theoretical results.

13.1 Introduction

In recent years the anaerobic digestion technology is widely used in biological wastewater treatment processes. This is due to its capacity for degrading highly concentrated organic substrates and at the same time for producing valuable energy (methane). The performance of these processes poses however a number of practical problems, since they are known to become easily unstable under parameter perturbations and variations of the operating conditions [1], [2]. To overcome this drawback, one needs control procedures to enhance the stable performance of

Neli S. Dimitrova
Institute of Mathematics and Informatics, Bulgarian Academy of Sciences,
Acad. G. Bonchev Str. 8, 1113 Sofia, Bulgaria
e-mail: nelid@bio.bas.bg

Mikhail I. Krastanov
Institute of Mathematics and Informatics, Bulgarian Academy of Sciences
Acad. G. Bonchev Str. 8, 1113 Sofia, Bulgaria
e-mail: krast@math.bas.bg

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_13,

the wastewater treatment plant. The first step in this direction is to have a practically validated dynamic model of the process. Using such models for control design and optimization has been proved to offer potential economic benefits. But even in this case, model-based control of anaerobic digestion is a complicated problem due to the difficulty in online monitoring the key biological variables, estimating the microorganisms growth rates, yield coefficients etc. Thus developing control systems only based on simple measurements that guarantee stability of the process is of primary importance. More information about different control approaches can be found in [2], [13], [18] and the references therein. Optimization via extremum (peek) seeking is another control approach extensively used in the last decade in order to optimize the productivity of a continuously stirred tank bioreactor [3], [19], [21], [22], [23]. In the literature, the extremum seeking approach is not model-based: the algorithm is usually presented in the form of a block-scheme to iteratively adjust the dilution rate directly in the bioreactor in order to steer the process to a point, where optimal value of the output is achieved. The main restriction in applying this model-free extremum seeking approach is that the dynamics should be open-loop stable. Otherwise, a locally stabilizing controller is needed to stabilize the equilibrium points around the optimal operating point. More details on the method can be found in [3], Chapters 5 and 8, as well as in [23].

The present paper continues the authors' investigations in [8], [9] on adaptive asymptotic stabilization of a four-dimensional nonlinear control system, which models an anaerobic biological wastewater treatment process. Here we propose a new feedback control law to stabilize asymptotically the closed-loop system towards an operating point, represented as a linear combination of the substrate concentrations. A model-based numerical extremum seeking algorithm is then designed and applied to stabilize the closed-loop system towards the equilibrium point with maximal methane output flow rate. The robustness of the algorithm is demonstrated by assuming uncertainties in the specific growth rate kinetics of the model.

The paper is organized as follows. Section 2 presents shortly the dynamic model of the wastewater treatment process and reports in more details on previous results by the authors related to adaptive stabilization of this model. The main and new result on asymptotic stabilization of the dynamic system towards a previously chosen operating point (called also reference or set point) is studied in Section 3. In order to prove that the closed-loop system is asymptotically stable, suitable Lyapunov-like functions are constructed explicitly. We show further in Section 4 how a numerical model-based extremum seeking algorithm can be used to stabilize the dynamic system towards the equilibrium point with maximal methane flow rate by choosing different operating points appropriately. Computer simulations illustrating the theoretical results are reported in Section 5. For convenience of the reader the extremum seeking algorithm is sketched in the Appendix.

Table 13.1: Definition of the model variables and parameters

s_1	concentration of chemical oxygen demand (COD) [g/l]
s_2	concentration of volatile fatty acids (VFA) [mmol/l]
x_1	concentration of acidogenic bacteria [g/l]
x_2	concentration of methanogenic bacteria [g/l]
u	dilution rate [day^{-1}]
s_1^i	influent concentration s_1 [g/l]
s_2^i	influent concentration s_2 [mmol/l]
k_1	yield coefficient for COD degradation [g COD/(g x_1)]
k_2	yield coefficient for VFA production [mmol VFA/(g x_1)]
k_3	yield coefficient for VFA consumption [mmol VFA/(g x_2)]
k_4	coefficient [l^2/g]
$\mu_{\max}$	maximum acidogenic biomass growth rate [day^{-1}]
μ_0	maximum methanogenic biomass growth rate [day^{-1}]
k_{s_1}	saturation parameter associated with s_1 [g COD/l]
k_{s_2}	saturation parameter associated with s_2 [mmol VFA/l]
k_I	inhibition constant associated with s_2 [(mmol VFA/l)$^{1/2}$]
α	proportion of dilution rate reflecting process heterogeneity
Q	methane gas flow rate

13.2 Model Description and Previous Results

We consider a model of an anaerobic digestion process, described by the following nonlinear system of ordinary differential equations [2], [12], [14]

$$\frac{ds_1}{dt} = u(s_1^i - s_1) - k_1\mu_1(s_1)x_1 \tag{13.1}$$

$$\frac{dx_1}{dt} = (\mu_1(s_1) - \alpha u)x_1 \tag{13.2}$$

$$\frac{ds_2}{dt} = u(s_2^i - s_2) + k_2\mu_1(s_1)x_1 - k_3\mu_2(s_2)x_2 \tag{13.3}$$

$$\frac{dx_2}{dt} = (\mu_2(s_2) - \alpha u)x_2 \tag{13.4}$$

$$Q = k_4\mu_2(s_2)x_2. \tag{13.5}$$

The state variables s_1, s_2 and x_1, x_2 denote substrate and biomass concentrations, respectively: s_1 represents the organic substrate, characterized by its chemical oxygen demand (COD), s_2 denotes the volatile fatty acids (VFA), x_1 and x_2 are the acidogenic and methanogenic bacteria respectively. The parameter $\alpha \in [0,1]$ represents the proportion of bacteria that are affected by the dilution; $\alpha = 0$ and $\alpha = 1$ correspond to an ideal fixed bed reactor and to an ideal continuous stirred tank reactor, respectively (cf. [1], [2], [4], [5], [12], [14], [20]).

The input substrate concentrations s_1^i and s_2^i are assumed to be constant. The dilution rate u is considered as a control input.

The definition of the model parameters is given in Table 1. There the constants μ_0, $\mu_{\max}$, k_{s_1}, k_{s_2} and k_I are related to the particular expressions of the specific growth rate functions $\mu_1(s_1)$ and $\mu_2(s_2)$, which are used later in Section 5.

Here we impose the following general assumptions on $\mu_1(s_1)$ and $\mu_2(s_2)$:

Assumption A1:
$\mu_i(s_i)$ is defined for $s_i \in [0,+\infty)$, $\mu_i(0) = 0$, $\mu_i(s_i) > 0$ for $s_i > 0$, $i = 1,2$;
$\mu_i(s_i)$ is continuously differentiable and bounded for all $s_i \in [0,+\infty)$, $i = 1,2$.

In a previous work [8] the authors design an adaptive stabilizing feedback control law for the same model in the presence of parameter uncertainties. This adaptive feedback depends on the observable state variables s_1 and x_1 and stabilizes asymptotically the closed-loop system towards an equilibrium point such that its projection on the s_1-axis is equal to a previously chosen operating point s_1^*.

The authors' investigations in [9] are based on the fact that the model (13.1)–(13.4) describes a two-stage process in a continuously stirred tank bioreactor [5], [14], based on two main reactions: (a) acidogenesis, where the organic substrate (denoted by s_1) is degraded into volatile fatty acids (VFA, denoted by s_2) by acidogenic bacteria (x_1); (b) methanogenesis, where VFA are degraded into methane CH_4 and carbon dioxide CO_2 by methanogenic bacteria (x_2). In it is shown in [9] that the open-loop system undergoes several local transcritical bifurcations of the steady states with respect to the control parameter u. This fact confirms the experimental observation that the dynamical open-loop system is highly unstable. Assuming that the acidogenesis (first stage, described by equations (13.1)–(13.2)) has been already stabilized to some operating point s_1^*, a nonlinear adaptive feedback is proposed in [9], which stabilizes asymptotically the closed-loop second stage dynamics (methanogenic phase) towards a previously chosen reference point $\overline{s}_2$, such that $(\overline{s}_2, \overline{x}_2)$ is an equilibrium point of (13.3)–(13.4). Further, a numerical extremum seeking algorithm is applied to steer the dynamics to an equilibrium point with maximum methane production. The robustness of the proposed feedback is demonstrated by assuming uncertainties in the model parameters.

Here we propose a new feedback law, that stabilizes simultaneously the whole system (13.1)–(13.4). The feedback depends on the so called biochemical oxygen demand (BOD), which can be represented as a linear combination of the substrate concentrations s_1 and s_2. For the practical application of the proposed feedback control it is worth to note that BOD is online measurable. This fact is discussed in details in [5]. The interested reader can find an overview of existing observers in [1] and [2]. Information about more specialized biosensors and numerical estimators is given in [6], [11] and the references therein.

13.3 Adaptive Asymptotic Stabilization

In this section we shall construct an adaptive stabilizing controller of (13.1)–(13.4). First we make the following technical assumption:

Assumption A2: The methane gas flow rate Q and the BOD concentration $\frac{k_2}{k_1}s_1 + s_2$ are online measurable.

Let us fix an operating (reference) point $\overline{s}$,

$$\overline{s} \in (0, s^i) \quad \text{with} \quad s^i := \frac{k_2}{k_1}s_1^i + s_2^i.$$

Assume that there exists a point $\overline{s}_1$ such that

$$\mu_1(\overline{s}_1) = \mu_2\left(\overline{s} - \frac{k_2}{k_1}\overline{s}_1\right), \quad \overline{s}_1 \in \left(0, s_1^i\right). \tag{13.6}$$

The above condition (13.6) is called regulability [12] of the system. Define further

$$\overline{s}_2 = \overline{s} - \frac{k_2}{k_1}\overline{s}_1, \quad \overline{x}_1 = \frac{s_1^i - \overline{s}_1}{\alpha k_1}, \quad \overline{x}_2 = \frac{s_2^i - \overline{s}_2 + \alpha k_2 \overline{x}_1}{\alpha k_3} = \frac{s^i - \overline{s}}{\alpha k_3}. \tag{13.7}$$

It is straightforward to see that the point

$$\overline{\zeta} := (\overline{s}_1, \overline{x}_1, \overline{s}_2, \overline{x}_2)$$

is an equilibrium point for the system (13.1)–(13.4). Our goal is to construct an adaptive feedback law to asymptotically stabilize the system (13.1)–(13.4) to $\overline{\zeta}$. Denote further by

$$Q(\overline{\zeta}) = k_4\, \mu(\overline{s}_2)\overline{x}_2 \tag{13.8}$$

the static characteristic of the model, which is defined on the set of all steady states. We shall also show that the adaptive feedback law can be applied so that to stabilize the control system (13.1)–(13.4) to an equilibrium point where the static characteristic of the model is maximal.

Denoting

$$s := \frac{k_2}{k_1}s_1 + s_2$$

we define the following sets

$$\begin{aligned}
\Omega_0 &= \{(s_1, x_1, s_2, x_2) |\, s_1 > 0, x_1 > 0, s_2 > 0, x_2 > 0\}, \\
\Omega_1 &= \left\{(s_1, x_1, s_2, x_2) |\, s_1 + k_1 x_1 \le \frac{s_1^i}{\alpha},\ s + k_3 x_2 \le \frac{s^i}{\alpha}\right\}, \\
\Omega_2 &= \left\{\left(s_1, x_1, \overline{s} - \frac{k_2}{k_1}s_1, \overline{x}_2\right) |\; 0 < s_1 < \frac{k_1}{k_2}\overline{s},\ x_1 > 0\right\} \\
\Omega &= \Omega_0 \cap \Omega_1.
\end{aligned}$$

Assumption A3: Let the inequality $\frac{d}{ds_1}\mu_1(s_1) + \frac{k_2}{k_1} \cdot \frac{d}{ds_1}\mu_2\left(\overline{s} - \frac{k_2}{k_1}s_1\right) > 0$ be satisfied on the set $\Omega \cap \Omega_2$.

Remark 13.1. Assumption A3 is technical and it is used in the proof of the main result. It is remarkable that this assumption is fulfilled whenever μ_1 and μ_2 are the Monod and the Haldane model functions and the values of the parameters are determined through off-line measurements (cf. [1], [2]).

The main result of this section is the following

Theorem 13.1. *Let us fix an arbitrary reference point* $\overline{s} \in (0, s^i)$. *Let Assumptions* A1*,* A2 *and* A3 *be satisfied. Then the control system* (13.1)–(13.4) *can be asymptotically stabilized to the point* $\overline{\zeta} = (\overline{s}_1, \overline{x}_1, \overline{s}_2, \overline{x}_2)$ *for each starting point* ζ_0 *from the set* Ω_0.

Proof. Let us fix an arbitrary point $\zeta_0 \in \Omega_0$ and a positive value $u_0 > 0$ for the control. According to Lemma 1 from [12] there exists $T > 0$ such that the value of the corresponding trajectory of (13.1)–(13.4) for $t = T$ belongs to the set Ω. Moreover, one can directly check that the set Ω is strongly invariant (cf. [7]) with respect to the trajectories of the control system (13.1)–(13.4). Hence the trajectory of (13.1)–(13.4) starting from the point ζ_0 enters the set Ω after a finite time and remains in Ω. For that reason we shall consider the control system (13.1)–(13.4) only on the set Ω.

Following [2], we extend the system (13.1)–(13.4) by adding the differential equation

$$\frac{d\beta}{dt} = -C(\beta - \beta^-)(\beta^+ - \beta)k_4\,\mu_2(s_2)\,x_2\,(s - \overline{s}), \tag{13.9}$$

where $C > 0$ is an arbitrary constant. Denote

$$\overline{\beta} = \frac{1}{\alpha k_4 \overline{x}_2} = \frac{k_3}{k_4(s^i - \overline{s})} \tag{13.10}$$

and let $\beta^- > 0$ and $\beta^+ > 0$ be arbitrary real numbers such that $\overline{\beta} \in (\beta^-, \beta^+)$. Consider the augmented set

$$\widetilde{\Omega} := \Omega \times (\beta^-, \beta^+);$$

with $\zeta := (s_1, x_1, s_2, x_2)$ define the following feedback control law

$$k(\zeta, \beta) := \beta\, k_4\, \mu_2(s_2)\, x_2, \quad (\zeta, \beta) \in \widetilde{\Omega}. \tag{13.11}$$

According to Assumption A2, the proposed feedback uses only online measurable quantities.

Consider the closed-loop system obtained from (13.1)–(13.4) and (13.9) by substituting the control variable u by the feedback $k(\zeta, \beta)$

$$\begin{aligned}
\frac{ds_1}{dt} &= k(\zeta,\beta)\cdot(s_1^i - s_1) - k_1\mu_1(s_1)x_1\\
\frac{dx_1}{dt} &= (\mu_1(s_1) - \alpha k(\zeta,\beta))x_1\\
\frac{ds_2}{dt} &= k(\zeta,\beta)(s_2^i - s_2) + k_2\mu_1(s_1)x_1 - k_3\mu_2(s_2)x_2\\
\frac{dx_2}{dt} &= (\mu_2(s_2) - \alpha k(\zeta,\beta))x_2\\
\frac{d\beta}{dt} &= -C(\beta-\beta^-)(\beta^+ - \beta)k_4\,\mu_2(s_2)\,x_2\,(s-\overline{s})\\
&\quad (\zeta_0,\beta_0) \in \widetilde{\Omega}.
\end{aligned} \tag{13.12}$$

For (13.12) we define the following function

$$V(\zeta,\beta) = (s-\overline{s}+k_3(x_2-\overline{x}_2))^2 +$$
$$\Gamma\left(\int_{\overline{s}}^{s}\frac{v-\overline{s}}{s^i - v}\,dv + \frac{1}{C}\int_{\overline{\beta}}^{\beta}\frac{w-\overline{\beta}}{(w-\beta^-)(\beta^+ - w)}\,dw\right),$$

where the parameter $\Gamma > 0$ will be determined later. Clearly, the values of this function are nonnegative. If we denote by $\dot{V}(\zeta,\beta)$ the Lie derivative of the function V with respect to the right-hand side of (13.12) at the point (ζ,β), then it can be directly checked that for each point (ζ,β) from the set $\widetilde{\Omega}$ the following equality holds true:

$$\begin{aligned}
\dot{V}(\zeta,\beta) = {} & -k(\zeta,\beta)\left(2+\Gamma\cdot\frac{k_3}{k_4\beta(s^i - s)(s^i-\overline{s}))}\right)(s-\overline{s})^2\\
&-2(1+\alpha)k_3\cdot k(\zeta,\beta)(s-\overline{s})(x_2-\overline{x}_2)\\
&-2\alpha k_3^2\cdot k(\zeta,\beta)(x_2-\overline{x}_2)^2.
\end{aligned}$$

The boundedness of the set $\widetilde{\Omega}$ implies the existence of a sufficiently large constant $\Gamma > 0$ so that

$$\dot{V}(\zeta,\beta) \le 0 \ \text{ for each point } \ (\zeta,\beta)\in\widetilde{\Omega}. \tag{13.13}$$

Let us denote by $\widetilde{\Omega}_2$ the set, where the Lie derivative of the function V with respect to the right-hand side of the closed system (13.12) is equal to zero. One can directly check that

$$\widetilde{\Omega}_2 := \left\{(s_1,x_1,s_2,\overline{x}_2,\overline{\beta})\in\widetilde{\Omega} : \frac{k_2}{k_1}s_1 + s_2 = \overline{s}\right\},$$

or equivalently

$$\widetilde{\Omega}_2 = \left\{\left(s_1,x_1,\overline{s}-\frac{k_2}{k_1}s_1,\overline{x}_2,\overline{\beta}\right)\in\widetilde{\Omega}\right\}.$$

Applying the LaSalle's invariance principle (cf. [16], Theorem 4.4), it follows that every solution of (13.12) starting from a point of $\widetilde{\Omega}$ is defined on the interval $[0,+\infty)$ and approaches the largest invariant set Ω_∞ (with respect to (13.12)) which is contained in the set $\widetilde{\Omega}_2$. In fact one can directly check that the set $\widetilde{\Omega}_2$ is invariant with respect to the trajectories of (13.12). Using (13.7), (13.10) and (13.11), the dynamics of (13.12) on the set $\widetilde{\Omega}_2$ can be described by the following system

$$\begin{aligned} \frac{ds_1}{dt} &= \frac{1}{\alpha}\chi(s_1)(s_1^i - s_1) - k_1\mu_1(s_1)x_1 \\ \frac{dx_1}{dt} &= (\mu_1(s_1) - \chi(s_1))x_1, \end{aligned} \tag{13.14}$$

where $\chi(s_1) := \mu_2\left(\bar{s} - \frac{k_2}{k_1}s_1\right)$. Obviously $\chi(s_1) > 0$ on the set $\widetilde{\Omega}_2$ due to Assumption 1. Taking into account that $\bar{s} = \frac{k_2}{k_1}\bar{s}_1 + \bar{s}_2$ and $s_1^i = \bar{s}_1 + \alpha k_1\bar{x}_1$, (13.14) can be rewritten as follows:

$$\begin{aligned} \frac{ds_1}{dt} &= -\frac{1}{\alpha}\chi(s_1)\cdot(s_1 - \bar{s}_1 + \alpha k_1(x_1 - \bar{x}_1)) - k_1\left(\mu_1(s_1) - \chi(s_1)\right)\cdot x_1 \\ \frac{dx_1}{dt} &= (\mu_1(s_1) - \chi(s_1))\cdot x_1. \end{aligned} \tag{13.15}$$

Consider the function

$$W(\zeta,\beta) = (s_1 - \bar{s}_1 + \alpha k_1(x_1 - \bar{x}_1))^2 + \alpha(1-\alpha)k_1^2(x_1 - \bar{x}_1)^2, \quad (\zeta,\beta) \in \widetilde{\Omega}_2.$$

This function takes nonnegative values and obviously depends only on s_1 and x_1, i. e. $W(\zeta,\beta) = W(s_1,x_1)$. Therefore, the Lie derivative $\dot{W}$ of W with respect to the right-hand side of (13.15) is presented in the following way:

$$\begin{aligned} \dot{W}(s_1,x_1) &= -\frac{2}{\alpha}\chi(s_1)(s_1 - \bar{s}_1 + \alpha k_1(x_1 - \bar{x}_1))^2 \\ &\quad - 2(1-\alpha)k_1x_1(s_1 - \bar{s}_1)(\mu_1(s_1) - \chi(s_1)). \end{aligned} \tag{13.16}$$

The regulability condition (13.6) implies the equality

$$\begin{aligned}\mu_1(s_1)-\chi(s_1) &= \mu_1(s_1)-\mu_2\left(\overline{s}-\frac{k_2}{k_1}s_1\right)\\ &= \mu_1(s_1)-\mu_2\left(\overline{s}_2-(s_1-\overline{s}_1)\frac{k_2}{k_1}\right)\\ &= \mu_1(\overline{s}_1)+\int_{\overline{s}_1}^{s_1}\frac{d\mu_1}{ds_1}(\theta)\,d\theta-\mu_2(\overline{s}_2)\\ &+\frac{k_2}{k_1}\int_{\overline{s}_1}^{s_1}\frac{d\mu_2}{ds_2}\left(\overline{s}_2-(\theta-\overline{s}_1)\frac{k_2}{k_1}\right)\,d\theta\\ &= \int_{\overline{s}_1}^{s_1}\left(\frac{d\mu_1}{ds_1}(\theta)+\frac{k_2}{k_1}\frac{d\mu_2}{ds_2}\left(\overline{s}_2-(\theta-\overline{s}_1)\frac{k_2}{k_1}\right)\right)\,d\theta.\end{aligned}$$

By means of Assumption A3 it follows that

$$(s_1-\overline{s}_1)\int_{\overline{s}_1}^{s_1}\left(\frac{d\mu_1}{ds_1}(\theta)+\frac{k_2}{k_1}\frac{d\mu_2}{ds_2}\left(\overline{s}_2-(\theta-\overline{s}_1)\frac{k_2}{k_1}\right)\right)\,d\theta>0.$$

From this inequality and from (13.16) we obtain that

$$\dot{W}(s_1,x_1)<0 \tag{13.17}$$

for each point $(s_1,x_1,\overline{s}-\frac{k_2}{k_1}s_1,\overline{x}_2,\overline{\beta})$ from the set $\widetilde{\Omega}_2\setminus\{(\overline{s}_1,\overline{x}_1,\overline{s}_2,\overline{x}_2,\overline{\beta})\}$.

To complete the proof we use an idea from [10] (cf. the proof of Theorem 3.1). First we shall remind some notions. Let us denote by $\phi(t,\zeta,\beta)$ the value of the trajectory of the closed-loop system (13.12) at time t starting from the point $(\zeta,\beta)\in\widetilde{\Omega}$. The positive limit set (or ω-limit set) of the solution $\phi(t,\zeta,\beta)$ of (13.12) is defined as

$$L^+(\zeta,\beta)=\Big\{(\tilde{\zeta},\tilde{\beta})|\text{ there exists a sequence }\{t_n\}\to+\infty\\ \text{with }\ (\tilde{\zeta},\tilde{\beta})=\lim_{t_n\to+\infty}\phi(t_n,\zeta,\beta)\Big\}.$$

The negative limit set (or α-limit set) $L^-(\zeta,\beta)$ of the solution $\phi(t,\zeta,\beta)$ of (13.12) is defined in an analogous way using sequences $\{t_n\}\to-\infty$.

Let us fix an arbitrary point (ζ_0,β_0) from the set $\widetilde{\Omega}$. The invariance of the bounded set $\widetilde{\Omega}$ with respect to the trajectories of (13.12) implies that the ω-limit set $L^+(\zeta_0,\beta_0)$ is a nonempty compact connected invariant set. Moreover, the LaSalle's invariance principle implies that $L^+(\zeta_0,\beta_0)$ is a subset of $\Omega_\infty\subseteq\widetilde{\Omega}_2$.

We shall prove that $L^+(\zeta_0,\beta_0)=\{(\overline{\zeta},\overline{\beta})\}$. Let us assume the contrary, i. e. there exists a point $(\zeta_\infty,\overline{\beta})\in L^+(\zeta_0,\beta_0)$ with $\zeta_\infty\neq\overline{\zeta}$. Then $\varepsilon:=\|(\zeta_\infty,\overline{\beta})-(\overline{\zeta},\overline{\beta})\|>0$.

The invariance of the set $L^+(\zeta_0,\beta_0)$ with respect to the trajectories of (13.12) implies that $\phi(-t,\zeta,\beta)\in L^+(\zeta_0,\beta_0)$ for each positive t and for each point $(\zeta,\beta)\in L^+(\zeta_0,\beta_0)$. In particular we have that $\phi(-t,\zeta_\infty,\overline{\beta})\in L^+(\zeta_0,\beta_0)$ for each positive t and hence $L^-(\zeta_\infty,\overline{\beta})\subseteq L^+(\zeta_0,\beta_0)$.

The inequality (13.17) implies the existence of a sequence $t_n\to+\infty$ such that

$$\lim_{t_n \to +\infty} \phi(t_n, \widehat{\zeta}, \overline{\beta}) = (\overline{\zeta}, \overline{\beta}), \text{ where } (\widehat{\zeta}, \overline{\beta}) \in L^-(\zeta_\infty, \overline{\beta}).$$

On the other hand, the invariance of the set $L^-(\zeta_\infty, \overline{\beta})$ with respect to the trajectories of (13.12) implies that each point $\phi(t_n, \widehat{\zeta}, \overline{\beta}) \in L^-(\zeta_\infty, \overline{\beta})$, $n = 1, 2, \ldots$. Then the closeness of the set $L^-(\zeta_\infty, \overline{\beta})$ implies that $\lim_{t_n \to +\infty} \phi(t_n, \widehat{\zeta}, \overline{\beta})$ also belongs to the set $L^-(\zeta_\infty, \overline{\beta})$. Thus we have obtained the following relation

$$(\overline{\zeta}, \overline{\beta}) \in L^-(\zeta_\infty, \overline{\beta}) \subseteq L^+(\zeta_0, \beta_0). \tag{13.18}$$

Let $B((\overline{\zeta}, \overline{\beta}), \varepsilon/3)$ be a closed ball centered at $(\overline{\zeta}, \overline{\beta})$ with radius $\varepsilon/3$. The first inclusion of (13.18) implies the existence of a sufficiently large number $T > 0$ such that $\phi(-T, \zeta_\infty, \overline{\beta}) = (\zeta_1, \beta_1) \in B((\overline{\zeta}, \overline{\beta}), \varepsilon/3)$. But this means that

$$\phi(T, \zeta_1, \beta_1) = (\zeta_\infty, \overline{\beta}). \tag{13.19}$$

The invariance of the set Ω_∞ with respect to the trajectories of (13.12) implies

$$(\zeta_1, \beta_1) \in B((\overline{\zeta}, \overline{\beta}), \varepsilon/3) \cap \Omega_\infty. \tag{13.20}$$

Then (13.17), (13.19) and (13.20) contradict to the equality $\|(\zeta_\infty, \overline{\beta}) - (\overline{\zeta}, \overline{\beta})\| = \varepsilon$. This contradiction shows that $(\overline{\zeta}, \overline{\beta}) = \Omega_\infty \cap L^+(\zeta_0, \beta_0)$ and completes the proof. □

13.4 Extremum Seeking

According to Assumption A2, the BOD concentration $s = \frac{k_2}{k_1} s_1 + s_2$ and the effluent methane flow rate Q are online measurable. Denote by $\overline{s} \in (0, s^i)$ some reference point and consider $\overline{\zeta} = (\overline{s}_1, \overline{x}_1, \overline{s}_2, \overline{x}_2)$ where $\overline{s}_1, \overline{x}_1, \overline{s}_2$ and $\overline{x}_2$ are computed according to (13.6) and (13.7). We assume that the static characteristic

$$Q(\overline{\zeta}) = k_4 \, \mu_2(\overline{s}_2) \overline{x}_2,$$

which is defined on the set of all steady states $\overline{\zeta}$ has a maximum at a unique steady state point

$$\zeta_{\max} = (s_1^m, x_1^m, s_2^m, x_2^m), \quad s_{\max} = \frac{k_2}{k_1} s_1^m + s_2^m \in (0, s^i),$$

that is $Q_{\max} = Q(\zeta_{\max})$.

Our goal now is to stabilize the dynamic system towards the (unknown) maximum methane flow rate $Q_{\max}$. For that purpose we write equation (13.9) in the form

$$\frac{d\beta}{dt}(t) = -C \cdot (\beta(t) - \beta^-) \cdot (\beta^+ - \beta(t)) \cdot Q(t) \cdot (s(t) - \bar{s}), \tag{13.21}$$

where $Q(t)$ denotes the methane flow rate measured at time t. It should be pointed out that not only $Q(t)$ but also all quantities in (13.21) are online measurable. Therefore, the values of its solution can be determined online as well. Since the solution of (13.21) depends on $\bar{s}$, we denote it by $\beta_{\bar{s}}(t)$, $t \in [0, +\infty)$. The last fact allows us to apply on-line the feedback control law

$$(s, Q, \beta_{\bar{s}}) \longmapsto k(s, Q, \beta_{\bar{s}}) = \beta_{\bar{s}} Q. \tag{13.22}$$

According to Theorem 1, this feedback will asymptotically stabilize the closed-loop system (13.12) to the point $(\overline{\zeta}, \overline{\beta}_{\bar{s}})$ with $\overline{\beta}_{\bar{s}} = \dfrac{k_3}{k_4(s^i - \bar{s})}$.

To stabilize the dynamics (13.12) towards $Q_{\max}$ by means of the feedback (13.22), we use the numerical iterative extremum seeking algorithm (see Appendix). The algorithm is based on the fact that Theorem 1 is valid for *any* reference point $\bar{s} \in (0, s^i)$. Thus we can construct a sequence of points $\bar{s}^{(1)}, \bar{s}^{(2)}, \ldots, \bar{s}^{(n)}, \ldots$ and generate in a proper way a sequence of values for the methane flow rate $Q^{(1)}$, $Q^{(2)}$, ..., $Q^{(n)}$, ..., which converges to $Q_{\max}$. The algorithm, which is first presented in [9] for a two-dimensional model, can easily be adapted for the model considered here. The algorithm is carried out in two stages: on Stage I, an interval $[S] = [S^-, S^+]$ is found such that $[S^-, S^+] \subset (0, s^i)$ and $s_{\max} \in [S^-, S^+]$; on Stage II, the interval $[S]$ is refined using an elimination procedure based on a Fibonacci search technique [15]. Stage II produces the final interval $[S^-_{\max}, S^+_{\max}]$ such that $[S^-_{\max}, S^+_{\max}] \subseteq [S^-, S^+] \subset (0, s^i)$, $s_{\max} \in [S^-_{\max}, S^+_{\max}]$ and $S^+_{\max} - S^-_{\max} \leq \varepsilon$, where the tolerance $\varepsilon > 0$ is assumed to be specified by the user.

13.5 Numerical Simulation

In the computer simulation we consider the Monod and the Haldane model functions for $\mu_1(s_1)$ and $\mu_2(s_2)$ respectively:

$$\mu_1(s_1) = \frac{\mu_{\max} s_1}{k_{s_1} + s_1}, \qquad \mu_2(s_2) = \frac{\mu_0 s_2}{k_{s_2} + s_2 + \left(\dfrac{s_2}{k_I}\right)^2}. \tag{13.23}$$

These functions are used in the original model, derived and studied in [1, 2, 4, 5, 12, 14]. Obviously, $\mu_1(s_1)$ and $\mu_2(s_2)$ satisfy Assumption A1. As it is well known, there is a point $\tilde{s}_2$ such that $\mu_2(s_2)$ achieves its maximum at $\tilde{s}_2 = k_I \sqrt{k_{s_2}}$. Moreover, Q also has a maximum at a unique steady state point.

The following values for the model coefficients are given in [1] and specified by practical experiments:

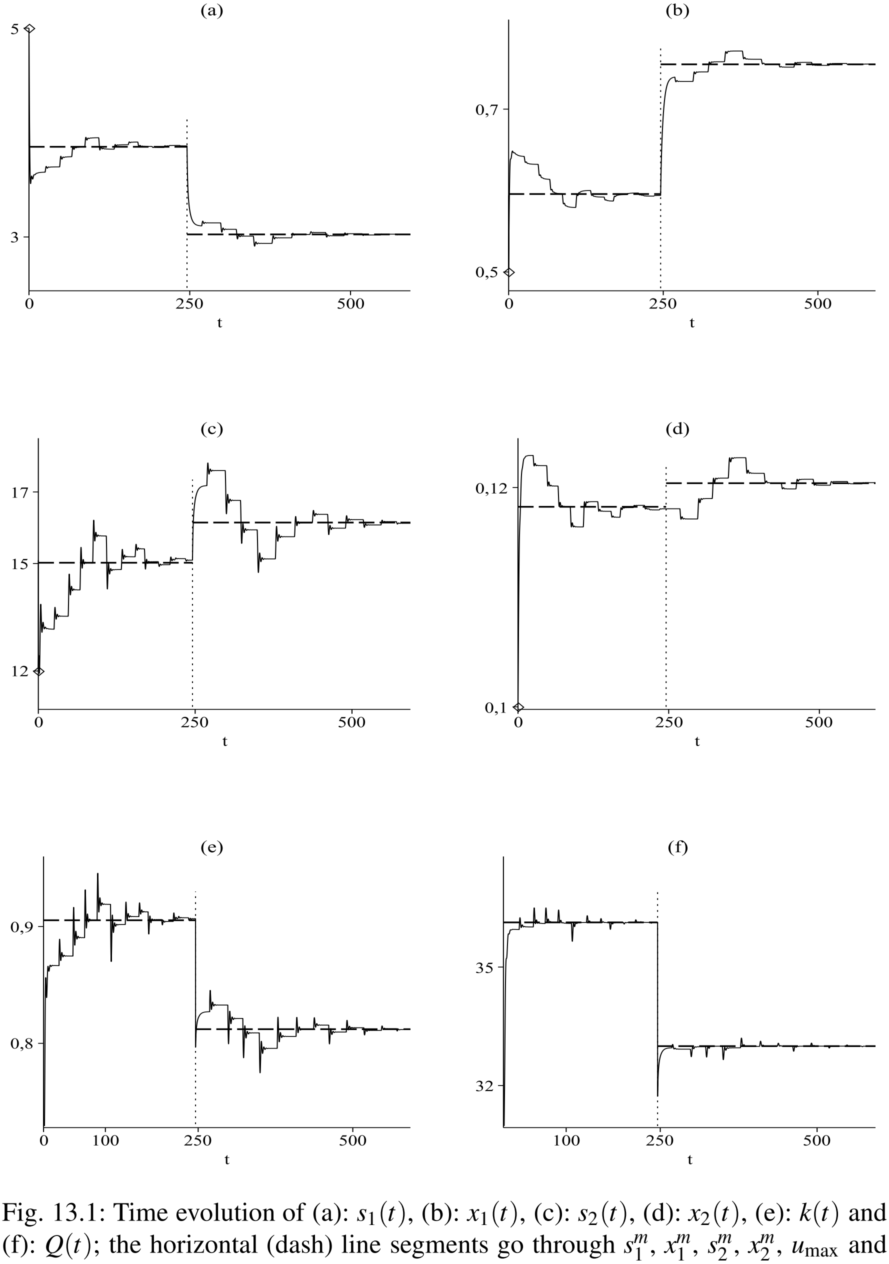


Fig. 13.1: Time evolution of (a): $s_1(t)$, (b): $x_1(t)$, (c): $s_2(t)$, (d): $x_2(t)$, (e): $k(t)$ and (f): $Q(t)$; the horizontal (dash) line segments go through s_1^m, x_1^m, s_2^m, x_2^m, $u_{\max}$ and $Q_{\max}$ respectively.

$$\begin{array}{llllll}
\alpha & = 0.5, & k_1 & = 10.53, & k_2 & = 28.6, \\
k_3 & = 1074, & k_4 & = 675 & \mu_{\max} & = 1.2, \\
k_{s_1} & = 7.1, & \mu_0 & = 0.74, & k_{s_2} & = 9.28, \\
k_I & = 16, & s_1^i & = 7, & s_2^i & = 70.
\end{array}$$

With the above coefficient values, the functions $\mu_1(s_1)$ and $\mu_2(s_2)$ satisfy Assumption A3.

Usually the formulation of the growth rates is based on experimental results, and therefore it is not possible to have an exact analytic form of these functions, but only some quantitative bounds. Assume that instead of $\mu_1(s_1)$ and $\mu_2(s_2)$ we know bounds for them, i. e.

$$\mu_i(s_i) \in [\mu_i(s_i)] = [\mu_i^-(s_i), \mu_i^+(s_i)] \text{ for all } s_i > 0, \quad i = 1,2.$$

This uncertainty can be simulated by assuming in (13.23) that instead of exact values for the kinetic coefficients $\mu_{\max}$, k_{s_1}, μ_0, k_{s_2} and k_I we have compact intervals for them:

$$\mu_{\max} \in [\mu_{\max}],\ k_{s_1} \in [k_{s_1}],\ \mu_0 \in [\mu_0],\ k_{s_2} \in [k_{s_2}],\ k_I \in [k_I].$$

Then any $\mu_i(s_i) \in [\mu_i(s_i)]$, $i = 1,2$, satisfies Assumption 1; in particular, the continuity of $\frac{d}{ds_i}\mu_i(s_i)$ implies the existence of intervals enclosing the above numerical values of the kinetic coefficients, such that Assumption 3 is also satisfied for any $\mu_i(s_i) \in [\mu_i(s_i)]$, $i = 1,2$. Such intervals are for example the following:

$$\begin{aligned} [\mu_{\max}] &= [1,\ 1.4], & [k_{s_1}] &= [6.5,\ 7.9], \\ [\mu_0] &= [0.64,\ 0.84], & [k_{s_2}] &= [8.28,\ 10.28], \quad [k_I] = [15,\ 17]. \end{aligned}$$

In the simulation process we proceed in the following way. At the initial time $t_0 = 0$ we take random values for the kinetic coefficients from the corresponding intervals. We apply the extremum seeking algorithm to stabilize the system (13.12) towards $Q_{\max}$. Then, at some time $t_1 > t_0$, we choose another set of random coefficient values and repeat the process; thereby the last computed values for the phase variables $(s_1, x_1, s_2, x_2, \beta)$ are taken as initial conditions.

The extremum seeking algorithm is implemented in the computer algebra system *Maple 13*. The standard ODE solver dsolve is used to solve the system (13.12) numerically. All intermediate numerical results are collected in arrays and then used to visualize the outputs.

Figure 1 shows the time profiles of the phase variables $s_1(t)$, $x_1(t)$, $s_2(t)$, $x_2(t)$ (plots (a) to (d) respectively), of the feedback $k(t)$ (plot (e)) and of $Q(t)$ (plot (f)). In the plots the symbol $\diamond$ denotes the initial values at $t_0 = 0$. The vertical dot line segments mark the time moment t_1, when the new coefficients values are taken in a random way from the corresponding intervals. The horizontal dash-line segments go through s_1^m, x_1^m, s_2^m, x_2^m, $u_{\max}$ and $Q_{\max}$ respectively, where $u_{\max} = k(\zeta_{\max}, \beta_{\max})$, $\beta_{\max} = \dfrac{k_3}{k_4(s^i - s_{\max})}$. The "jumps" in the graphs correspond to the different choices of $\bar{s}$ by executing the algorithmic steps.

The extremum seeking algorithm could be implemented to work online. In this case the choice of $\bar{s}$ will be determined from the measurements of BOD and Q.

13.6 Conclusion

The paper is devoted to the stabilization of a four-dimensional nonlinear dynamic system which models an anaerobic degradation of organic wastes and produces methane. A nonlinear adaptive feedback is proposed which stabilizes asymptotically the dynamic system towards the (unknown) maximum methane production rate $Q_{\max}$. For that purpose, it is first shown that for any previously chosen reference point $\bar{s}$ representing the biochemical oxygen demand the system can be asymptotically stabilized to an equilibrium point $(\bar{s}_1, \bar{x}_1, \bar{s}_2, \bar{x}_2)$, such that $\bar{s} = \frac{k_2}{k_1}\bar{s}_1 + \bar{s}_2$. Further, an iterative numerical extremum seeking algorithm is applied to stabilize *in real time* the closed-loop system into an interval $[S_{\max}]$ containing the equilibrium point $s_{\max}$ for which the methane flow rate Q takes its maximum $Q_{\max}$. The interval $[S_{\max}]$ can be made as tight as desired depending on a user predefined tolerance $\varepsilon > 0$. The robustness of the feedback as well as of the extremum seeking algorithm are demonstrated numerically by assuming that the coefficients in the expressions of the growth rate functions are not exactly known but bounded by compact intervals.

Appendix: The Extremum Seeking Algorithm

We present below the main steps of the numerical extremum seeking algorithm. The steps are executed in the given order except as indicated by branching. We assume tolerances $\varepsilon > 0$, $h > 0$ and $\varepsilon_s > 0$ to be given.

I. Determine an interval $[S] = [S^-, S^+]$ such that $[S] \subset (0, s^i)$ and $\bar{s}_{\max} \in [S]$.

Step I.0. Choose $\bar{s}^0 \in (0, s^i)$. Apply the feedback $k(s, Q, \beta_{\bar{s}^0})$ to stabilize the system to $\bar{s}^0$. According to Theorem 1, there exists a moment of time $t_0 > 0$ such that $|s(t_0) - \bar{s}^0| < \varepsilon_s$; set $\bar{s}^0 := s(t_0)$, $Q_0 := Q(t_0)$.

Step I.1. Set $\sigma := 1$, $\bar{s}^1 := \bar{s}^0 + \sigma h$. Apply the feedback $k(s, Q, \beta_{\bar{s}^1})$ to stabilize the system to $\bar{s}^1$. According to Theorem 1, there exists a moment of time $t_1 > 0$ such that $|s(t_1) - \bar{s}^1| < \varepsilon_s$; set $\bar{s}^1 := s(t_1)$, $Q_1 := Q(t_1)$. If $Q_1 > Q_0$ then goto Step I.3 else goto Step I.2.

Step I.2. Set $\sigma := -1$, $\bar{s}^1 := \bar{s}^0 + \sigma h$. Apply the feedback $k(s, Q, \beta_{\bar{s}^1})$ to stabilize the system to $\bar{s}^1$. According to Theorem 1, there exists a moment of time $t_1 > 0$ such that $|s(t_1) - \bar{s}^1| < \varepsilon_s$; set $\bar{s}^1 := s(t_1)$, $Q_1 := Q(t_1)$.

```
    If $Q_1 > Q_0$ then goto Step I.3.
    If $Q_1 \le Q_0$ then set $h := h/2$;
      if $h \le \varepsilon/2$ then set $[S_{\max}] := [\bar{s}^0 - \varepsilon, \bar{s}^0 + \varepsilon]$;   go to III;
      if $h > \varepsilon/2$ then goto Step I.1.
```

Step I.3. Set $h := 2h$, $\bar{s}^2 := \bar{s}^1 + \sigma h$. Apply the feedback $k(s,Q,\beta_{\bar{s}^2})$ to stabilize the system to $\bar{s}^2$. According to Theorem 1, there exists a moment of time $t_2 > 0$ such that $|s(t_2) - \bar{s}^2| < \varepsilon_s$; set $\bar{s}^2 := s(t_2)$, $Q_2 := Q(t_2)$.

If $Q_2 \leq Q_1$ then set $[S] = [S^-, S^+] := [\bar{s}^0, \bar{s}^2]$ and goto *II.*

If $Q_2 > Q_1$ then set $\bar{s}^0 := \bar{s}^1$, $\bar{s}^1 := \bar{s}^2$, $Q_1 := Q_2$; repeat this Step I.3.

II. Starting with $[S] = [S^-, S^+]$, determine an interval $[S_{\max}] = [S^-_{\max}, S^+_{\max}]$ with $s_{\max} \in [S_{\max}]$ and $S^+_{\max} - S^-_{\max} \leq \varepsilon$.

Denote $\bar{s}^{0^-} := S^-$, $\bar{s}^{0^+} := S^+$, $\lambda := \dfrac{\sqrt{5}-1}{2}$; compute $\Delta_1 := \bar{s}^{0^+} - \bar{s}^{0^-}$.

Step II.0. Compute $\Delta_2 := (1-\lambda)\Delta_1$, $p_0 := \bar{s}^{0^-} + \Delta_2$, $q_0 := \bar{s}^{0^+} - \Delta_2$.

Step II.1. Apply the feedback $k(s,Q,\beta_{p_0})$ to stabilize the system to p_0. According to Theorem 1, there exists a moment of time $t_{p_0} > 0$ such that $|s(t_{p_0}) - p_0| < \varepsilon_s$; set $p_0 := s(t_{p_0})$, $Q_{p_0} := Q(t_{p_0})$.

Apply the feedback $k(s,Q,\beta_{q_0})$ to stabilize the system to q_0. According to Theorem 1, there exists a moment of time $t_{q_0} > 0$ such that $|s(t_{q_0}) - q_0| < \varepsilon_s$; set $q_0 := s(t_{q_0})$, $Q_{q_0} := Q(t_{q_0})$.

Step II.2. Set $\Delta_3 := q_0 - p_0$.

If $Q_{p_0} > Q_{q_0}$ then set $\bar{s}^{1^-} := \bar{s}^{0^-}$, $\bar{s}^{1^+} := q_0$, $p_1 := \bar{s}^{1^-} + \Delta_3$, $q_1 := p_0$;

If $Q_{p_0} \leq Q_{q_0}$ then set $\bar{s}^{1^-} := p_0$, $\bar{s}^{1^+} := \bar{s}^{0^+}$, $p_1 := q_0$, $q_1 := \bar{s}^{1^+} - \Delta_3$.

Compute $\Delta_1 := \bar{s}^{1^+} - \bar{s}^{1^-}$.

Step II.3.
```
If $\Delta_1 \leq \varepsilon$ then set $[S_{\max}] := [\bar{s}^{1^-}, \bar{s}^{1^+}]$;  goto III.
If $\Delta_1 > \varepsilon$ then
  if $p_1 \geq q_1$ then set $\bar{s}^{0^-} := \bar{s}^{1^-}$, $\bar{s}^{0^+} := \bar{s}^{1^+}$ and goto Step II.0.
  if $p_1 < q_1$ then
    if $Q_{p_0} > Q_{q_0}$ then apply the feedback $k(s,Q,\beta_{p_1})$ to stabilize the system to $p_1$. According to Theorem 1, there exists a moment of time $t_{p_1} > 0$ such that $|s(t_{p_1}) - p_1| < \varepsilon_s$;
    set $p_1 := s(t_{p_1})$, $Q_{p_1} := Q(t_{p_1})$.
    if $Q_{p_0} \leq Q_{q_0}$ then apply the feedback $k(s,Q,\beta_{q_1})$ to stabilize the system to $q_1$. According to Theorem 1, there exists a moment of time $t_{q_1} > 0$ such that $|s(t_{q_1}) - q_1| < \varepsilon_s$;
    set $q_1 := s(t_{q_1})$, $Q_{q_1} := Q(t_{q_1})$.
Set $p_0 := p_1$, $q_0 := q_1$, $\bar{s}^{0^-} := \bar{s}^{1^-}$, $\bar{s}^{0^+} := \bar{s}^{1^+}$,
$Q_{p_0} := Q_{p_1}$, $Q_{q_0} := Q_{q_1}$;
goto Step II.2.
```

III. STOP computations

Remark 13.2. At any step of the algorithm, the last computed values for s_1, x_1, s_2 and x_2 are used as initial conditions for the next step. For β, the last computed value is checked whether $\beta \in (\beta^-, \beta^+)$; if not, then it is changed to $\beta = (\beta^- + \beta^+)/2$.

Acknowledgements The authors are grateful to the anonymous referees for the valuable advices and comments.

This work has been partially supported by the Bulgarian Science Fund under grant No. DO 02–359/2008.

References

1. Alcaraz-González, V., Harmand, J., Rapaport A., Steyer, J.-P., González-Alvarez, V., Pelayo-Ortiz, C.: Software sensors for highly uncertain WWTPs: a new apprach based on interval observers. Water Research **36**, 2515–2524 (2002)
2. Antonelli, R., Harmand, J., Steyer, J.-P., Astolfi, A.: Set-point regulation of an anaerobic digestion process with bounded output feedback. IEEE Trans. Control Systems Tech. **11** (4), 495–504 (2003)
3. Ariyur, K. B., Ktstić, M.: Real-time optimization by extremum-seeking control. John Wiley & Sons, Hoboken, New Jersey (2003)
4. Bernard, O., Hadj-Sadok, Z., Dochain, D.: Advanced monitoing and control of anaerobic wastewate treatment plants: dynamic model development and identification. In: Proceedings of Fifth IWA Inter. Symp. WATERMATEX, Gent, Belgium, pp. 3.57–3.64 (2000)
5. Bernard, O., Hadj-Sadok, Z., Dochain, D., Genovesi, A., Steyer, J.-P.: Dynamical model development and parameter identification for an anaerobic wastewater treatment process. Biotechnology and Bioengineering **75**, 424-438 (2001)
6. Chang, I., Jang, K., Gil, G., Kim, M., Kim, H. Cho, B., Kim, B.: Continuous determination of biochemical oxygen demand using microbial fuel cell type biosensor. Biosensors and Bioelectronics **19**, 607-813 (2004)
7. Clarke, F., Ledyaev, Yu., Stern, R., Wolenski, P.: Nonsmooth Analysis and Control Theory. Graduate Text in Mathematics **178**, Springer, Berlin (1998)
8. Dimitrova, N., Krastanov, M. I.: Nonlinear adaptive control of an uncertain wastewater treatment model. In: IEEE Proceedings of the 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics, Duisburg, Germany, September 26-29, 2006, ISBN-13: 978-0-7695-2821-2, ISBN-10: 0-7695-2821-X, accessible through IEEE data base Xplore (2007)
9. Dimitrova, N., Krastanov, M. I.: Nonlinear stabilizing control of an uncertain bioprocess model. Int. Journ. of Appl. Math. Comput. Sci. **19.3**, 441–454 (2009)
10. El-Hawwary, M. I., Maggiore, M.: Reduction Principles and the Stabilization of Closed Sets for Passive Systems. IEEE Trans. Automatic Control **55** (4), 982–987 (2010)
11. Farza, M., Busawon, K., Hammouri, H.: Simple nonlinear observers for on-line estimation of kinetic rates in bioreactors. Automatica **34**, 301-318 (1998)
12. Grognard, F., Bernard, O.: Stability analysis of a wastewater treatment plant with saturated control. Water Science Technology **53**, 149-157 (2006)
13. Heinzle, E., Dunn, I., Ryhiner, G.: Modelling and control for anaerobic wastewater treatment. Advances in Biochemical Engineering and Biotechnology **48**, 79–114 (1993)
14. Hess, J., Bernard, O.: Design and study of a risk management criterion for an unstable anaerobic wastewater treatment process. Journal of Process Control **18** (1), 71–79 (2008)
15. Karmanov, V. Mathematical Programming. FIZMATLIT, Moskva (2000) (in Russian)
16. Khalil, H.: Nonlinear Systems. Macmillan Publishing Company, New York (1992)

17. Maillert, L., Bernard, O., Steyer, J.-P.: Robust regulation of anaerobic digestion processes. Water Science and Technology **48** (6), 87–94 (2003)
18. Maillert, L., Bernard, O., Steyer, J.-P.: Nonlinear adaptive control for bioreactors with unknown kinetics. Automatica **40**, 1379–1385 (2004)
19. Marcos, N. I., Guay, M., Dochain, D., Zhang, T.: Adaptive extremum-seeking control of a continuous stirred tank bioreactor with Haldane's kinetics. Journ. Process Control **14** (3), 317–328 (2004)
20. Schoefs, O., Dochain, D., Fibrianto, H., Steyer, J.-P.: Modelling and identification of a distributed-parameter system for an anaerobic wastewater treatment process. Chemical Eng. Research and Design. **81**.(A9), 1279–1288 (2003)
21. Simeonov, I., Noykova, N., Stoyanov, S.: Modelling and extremum seeking control of the anaerobic digestion. In: Proceedings of the International IFAC Workshop DECOM–TT, Bansko, Bulgaria, pp. 289–294 (2004)
22. Simeonov, I., Noykova, N., Gyllenberg, M.: Identification and extremum seeking control of the anaerobic digestion of organic wastes. Cybernetics and Information technologies **7** (2), 73–84 (2007)
23. Wang, H.-H., Krstić, M., Bastin, G.: Optimizing bioreactors by extremum seeking. Int. J. Adapt. Control Signal Process **13**, 651–669 (1999)

Chapter 14
Verified Analysis of a Model for Stance Stabilization

Ekaterina Auer (✉), Haider Albassam, Andrés Kecskeméthy and Wolfram Luther

Abstract The stabilization of stance is a subject of continuing research in biology, biomechanics and robotics. It plays an important role in many clinical applications as well as in forward dynamical gait simulation. In this paper, we propose a new model relying on a two cylinder foot contact scheme. This contact model has the advantage of simple and smooth dynamic behavior which in turn results in better efficiency in comparison with other contact models. However, a number of parameters in this model, such as position or mass of the pelvis, are known only with some uncertainty. To deal with the situation, we analyze the model using verified methods, which includes propagating the uncertainty through the system and computing the sensitivities of the equations of motion in the first time interval. To perform verified simulations of the whole model, a verified initial value problem solver for a hybrid system is required, which can switch from one system of the equations of motion to the other depending on a certain switching function. While research in this direction remains a topic of high complexity, a simplified kinetostatic version of the model allows one to analyze the sensitivity of the model to parameter variations, as presented in this paper.

Ekaterina Auer
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: auer@inf.uni-due.de

Haider Albassam
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: haider.albassam@uni-due.de

Andrés Kecskeméthy
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: andres.kecskemethy@uni-due.de

Wolfram Luther
University of Duisburg-Essen, D-47048 Duisburg, Germany
e-mail: luther@inf.uni-due.de

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_14,

14.1 Introduction

Biomechanics is a rapidly expanding area of research concerned with applying the principles of mechanics to biological systems mainly to solve medical problems. In the meanwhile, there exist many biomechanical simulation tools which are utilized in real life surgeries, as for example the total hip replacement planning tools [4], [15]. However, the classical approach has several drawbacks. For example, surgeons often have to use 2D images for 3D reconstruction, or they need to employ scaled bone and muscle models that match only roughly the individual patient data. The goal of the software *MobileBody* [18], [25], a diagnose program for human musculoskeletal system built on the basis of the multibody modeling and simulation software MOBILE [12], was to overcome these problems. It combines information gathered in the gait lab using a marker-based technology with MRT (magnetic resonance tomography) and Xray recordings into a patient-specific mechanical model.

In this setting, there are several subtasks that need to be solved, of which we consider here the problem of human stance stabilization. This subproblem plays an important role in forward dynamical gait simulation, which again is of importance for a number of clinical evaluations. Some parameters of the task, such as position or mass of the pelvis, are influenced by uncertainty. In our case, we have to deal with the so-called epistemic (reducible) type of uncertainty. This type comprises the incertitude due to lack of knowledge, an example of which is the absence of evidence about the probability distribution of a parameter. In the biomechanical case, such parameters are the lengths and masses of bones as well as their positions, which cannot be measured exactly.

One possibility to describe the imprecision in the outcome is to provide bounds enclosing all possible results (if the uncertain parameters are bounded). For this possibility, a range of tools is offered by the program SmartMOBILE [1], an extension of MOBILE. For example, SmartMOBILE allows the user to compute an enclosure for the length of the femur bone given the measuring uncertainties in the positions of markers attached to a human leg in order to identify the bone segment motion. SmartMOBILE uses verified techniques for this kind of tasks. Such methods guarantee the correctness of the outcome of a computer simulation using mathematically exact proofs based, for example, on fixed point theorems. Interval [19], Taylor model [17] or affine arithmetic [6] based methods are most prominent examples of verified techniques. Besides proving the correctness of the computed result, verified methods can take care of rounding errors and propagate bounded uncertainties through systems.

In this paper, we introduce a model for foot contact playing an important role in the process of stabilization of human stance. Although there exist several known models for this task [9], [11], [23], the proposed approach has the advantage of providing a simple model with smooth dynamic behavior which in turn results in better efficiency [25]. To analyze the influence of parameter uncertainty on the model, we make use of interval arithmetic to obtain verified bounds on the outcome and to compute parameter sensitivities. In order to do that for the whole model, a verified version of an initial value problem (IVP) solver for hybrid systems is required. That is, we have to deal with a system of ordinary differential equations (ODE) changing its right side depending on a certain choice function. To our knowledge, there is no such verified solver at present. Therefore, we reduce our analysis to investigating the influence of the uncertainty on the equations of motion in the first time interval. In particular, we compute verified bounds on the corresponding forces along with their interval and nominal sensitivities to a number of uncertain parameters.

The paper is structured as follows. In Section 14.2 we give a brief overview of verified methods, the tool MOBILE and its verified version SMARTMOBILE. In the next section, we describe the problem of human stance stabilization and the corresponding biomechanical model including its simplification. In Section 14.4 we report on our first analysis of parameter uncertainty influence on this model from the verified perspective. Finally, we recapitulate the main results and point out our future work in the last section.

14.2 Background

To be able to consider uncertainties in the model of human stance stabilization, we use SMARTMOBILE, a verified modeling and simulation tool based on MOBILE. In this section, we describe briefly the theory and libraries which make verified computations in SMARTMOBILE possible and give a short overview of the involved multibody software.

14.2.1 Verified Methods and Libraries

To model and simulate stabilization of human stance, we rely on interval arithmetic [19] in our first verified analysis. An interval $[\underline{x},\overline{x}]$, where $\underline{x}$ is the lower, $\overline{x}$ the upper bound, is defined as

$$X = [x] = [\underline{x},\overline{x}] = \{x \in \mathbb{R} | \underline{x} \leq x \leq \overline{x}\}.$$

Elementary operations and functions can be defined on intervals in such a way as to result in intervals. To be able to work with this definition on a computer using a finite precision arithmetic, the concept of machine intervals is necessary. Machine intervals are represented by floating point numbers for the lower and upper bounds. To obtain the corresponding machine interval for the real interval $[\underline{x},\overline{x}]$, the lower bound is rounded down to the largest representable machine number equal or less than $\underline{x}$, and the upper bound is rounded up to the smallest machine number equal or greater than $\overline{x}$. These notions can be extended to define interval vectors and matrices.

There exist interval analogs to higher-level numerical algorithms such as those for solving linear, nonlinear or differential systems of equations. The usual algorithms are reformulated in such a way as to guarantee the correctness of the computed outcome. That means that the enclosure they produce is proven to contain the exact result. Almost all algorithms need at least one derivative of the right side of system model equations to be able to work. That is, it is necessary to obtain derivatives of code automatically [10]. There are several libraries implementing this concept which employ either overloading or code transformation.

There also exist further verified methods, for example, affine arithmetic [6] or Taylor models [17]. They try to overcome one methodical difficulty always present in interval computations, namely, the dependency problem. According to the principles of interval computations, the two variables X, for example, in the expression $X - X$, are not considered to be the same (and therefore dependent) but rather treated independently. That is, we actually work with the expression $X - Y$. This is a source of considerable overestimation (too pessimistic bounds for the result, as in $X - X \neq [0,0]$) in interval arithmetic.

Another important concept for this paper is the sensitivity. We understand it as a linear measure of uncertainty influence. If we have a bounded uncertain parameter $[p]$, which our characteristic of interest $[x]$ depend on, then the sensitivity is defined as $[s] = \partial[x]/\partial[p]$. If this definition does not produce a meaningful result in interval arithmetic, we might use a reference from engineering:

$$[r] = \sum_i |\partial x/\partial p_i| \cdot [p_i] \tag{14.1}$$

(with interval operations). Here, x and p_i are reference values, for example, midpoints of the uncertain quantities.

There is a number of software libraries implementing this theory in different programming languages such as C++ or FORTRAN and computer algebra packages such as MAPLE or MATLAB. In SMARTMOBILE, we use PROFIL/BIAS [14] for basic interval operations and FADBAD++ [24] for algorithmic differentiation in this paper.

14.2.2 Piecewise Continuous Functions

Many functions which are to have physical meaning are in fact only piecewise continuous. For example, a normal contact force should be negative, which is usually expressed as an ”if-then-else” condition in code. Such characteristics are difficult from the point of view of algorithmic differentiation because they are differentiable only piecewise. There are several libraries, for example CPPAD [2], offering their own versions of ”if-then-else” conditions so as to get valid derivatives of the corresponding functions. However, they only work pointwise, that is, they cannot be applied if their arguments are proper intervals. To deal with this problem, we implemented a class `pwFunc` for computation of enclosures and first derivatives for piecewise functions for interval arguments. Let the piecewise function be defined in the following way:

$$f(x) = \begin{cases} f_0(x), & \text{if } c_{-1} = -\infty < x \le c_0, \\ f_1(x), & \text{if } c_0 < x \le c_1, \\ \dots & \dots \\ f_{n-1}(x), & \text{if } c_{n-2} < x \le c_{n-1}, \\ f_n(x), & \text{if } c_{n-1} < x < c_n = +\infty \end{cases}, \text{ where } c_i \text{ are constants.} \tag{14.2}$$

For such functions, we define an interval extension by

$$f(X) = \begin{cases} f_i(X), & \text{if } X \subset (c_{i-1}, c_i], \\ f_i([\underline{x}, c_i]) \cup \bigcup\limits_{k=i+1}^{j-1} f_k([c_{k-1}, c_k]) \cup f_j([c_{j-1}, \overline{x}]), & \text{if } X \subset (c_{i-1}, c_j] \end{cases}, \tag{14.3}$$

where $0 \le i < j \le n$, and an interval extension of the first derivative analogously as

$$f'(X) = \begin{cases} f_i'(X), & \text{if } X \subset (c_{i-1}, c_i], \\ f_i'([\underline{x}, c_i]) \cup \bigcup\limits_{k=i+1}^{j-1} f_k'([c_{k-1}, c_k]) \cup f_j'([c_{j-1}, \overline{x}]), & \text{if } X \subset (c_{i-1}, c_j] \end{cases}. \tag{14.4}$$

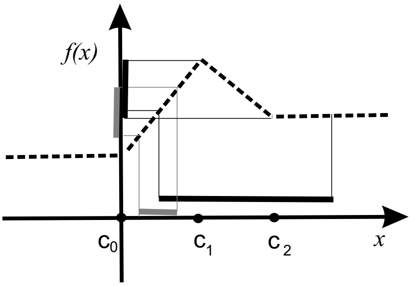


Fig. 14.1: An interval extension of a non-smooth function according to (14.3).

14.2.3 MOBILE and SmartMOBILE

MOBILE is an object oriented C++ environment for modeling and simulation of kinematics and dynamics of mechanical systems based on the multibody modeling method. Its central concept is to use as building blocks kinetostatic transmission elements which map motion and force between system states. For example, an elementary joint modeling revolute and prismatic joints is such a transmission element. Mechanical systems are considered to be concatenations of these entities. In this way, serial chains, tree type or closed-loop systems can be modeled. With the help of the global kinematics, the transmission function of the complete system chain can be obtained from transmission functions of its parts. The inverse kinematics and the kinetostatic method [12] help to build dynamic equations of motion, which are solved with common IVP solvers. MOBILE belongs to the numerical type of modeling software, that is, it does not produce a symbolic description of the resulting model. Only the values of output parameters for the user-defined values of input parameters and the source code of the program itself are available. In this case, it is necessary to integrate verified techniques into the core of the software itself, as opposed to the tools of the symbolical type, where the task is basically reduced to the application of the verified methods to the obtained system of equations.

All transmission elements in MOBILE are derived from the abstract class `MoMap`, which supplies their main functionality including the methods `doMotion()` and `doForce()` for transmission of motion and force. For example, elementary joints are modeled by the class `MoElementaryJoint`. Besides, there exist elements for modeling mass properties and applied forces. Transmission elements are assembled to chains implemented by the class `MoMapChain`. The methods `doMotion()` and `doForce()` can be used for a chain representing the system to determine the corresponding composite transmission function. The class `MoEqmBuilder` is responsible for generation of equations of motion, which are subsequently transferred into their state-space form by `MoMechanicalSystem`. Finally, the corresponding IVP is solved by an appropriate integrator algorithm, for example, Runge-Kutta's using the class `MoRungeKuttaIntegrator` derived from the basic class `MoIntegrator`.

SmartMOBILE (based on MOBILE) is one of the first integrated environments providing result verification for kinematic and dynamic simulations of mechanical systems. Models in both tools are executable C++ programs built of the supplied classes for transmission elements and solvers. The advantage of SmartMOBILE is its flexibility due to the template structure: the user can choose the kind of (non)verified arithmetics according to his task. Advanced users are not limited to the already defined classes for these arithmetics and are free to plug in their own implementations.

An overview of arithmetics available in SmartMOBILE at this moment is given in Table 14.1. For most kinematical problems, it is sufficient to use a basic data type from Column 3 of the Table 14.1 as the parameter of all the template classes used for a particular model. The main idea for dynamical and special kinematical tasks such as finding system equilibria is to use semantic pairs consisting of basic data type and corresponding solvers (Columns 3 and 4). Our experience shows that the general tendency as to what kind of arithmetic to use is as follows. If only a reference solution is of interest, floating point arithmetics with `MoReal` and a usual numerical integrator such as Runge-Kutta's can be employed for dynamic simulations. If the user is interested in fast verification of a relatively simple system with little uncertainty, interval-based pairs are of use. Taylor arithmetics should be mostly chosen for offline simulations with considerable uncertainty [1].

Besides verified modeling and simulation, SmartMOBILE offers techniques for sensitivity analysis and uncertainty management.

Table 14.1: Arithmetics supplied with SMARTMOBILE.

Description	Arithmetic	Kinematics	Dynamics
reference	floating point	MoReal	MoRungeKutta,...
based on VNODE [20]	intervals	TMoInterval	TMoAWA
based on VALENCIA-IVP [22]	intervals	TMoFInterval	TMoValencia
based on RIOT [5]	Taylor	TMoTaylorModel	TMoRiOT
based on COSY [17]	Taylor	RDAInterval	—
equilibrium states	intervals	MoFInterval	MoIGradient
sensitivity with VALENCIA-IVP	intervals	MoSInterval	TMoValenciaS

14.3 A Model for Human Stance Stabilization

The problem of modeling the stance can be divided into three stages [16]. First, human skeleton has to modeled. Our model consists of nine segments (cf. Figure 14.2): the pelvis representing the whole upper body, then right and left femur, right and left tibia as well as right and left foot composed of a forefoot and hindfoot each. These segments are connected by appropriate joints. The second, most important stage is the modeling of the foot contact. It is achieved by choosing two cylinders as contact surfaces for the foot and using a Hunt-Crossley contact scheme. Finally, a PID (proportional, integral and derivative) controller is applied to stabilize the stance.

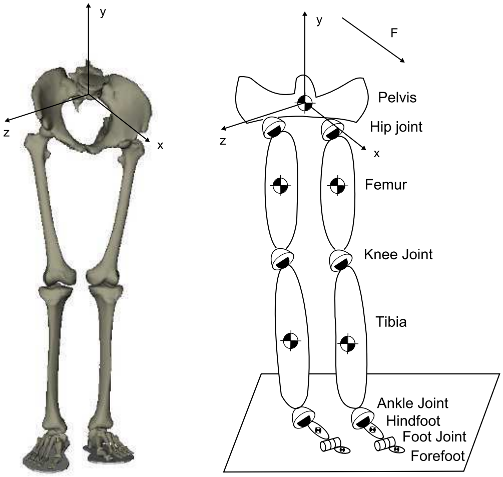


Fig. 14.2: Nine segment model of human skeleton.

In this section, we describe the foot contact model in detail and set out a possible simplification used afterwards for the verified model analysis.

14.3.1 Two-Cylinder Foot Contact Model

The human foot is a complex anatomical structure that has recently been intensively examined by the biomechanics community [9], [11], [23]. For example, a two-segment model with a single matatarsal-phalangeal joint was proposed in [9]. Our model consists of two rigid segments connected as in [9] by a revolute joint the axis of which is perpendicular to the sagittal plane in the neutral null position (see Figure 14.3). The first segment is attached to the hindfoot (Calcaneus), and the second to the forefoot (Phalanges). Additionally, two torsional spring-damper elements are fastened to the joint.

The reaction force of the impact between the two segments and the ground can be modeled as a function of the penetration. In the normal direction, the force can be described using the model of Hunt-Crossley:

$$F_N = C_N x_N^n \left(1 + \frac{2}{3}\frac{1}{v_N^-}\left(\frac{1}{e_N} - 1\right)\dot{x}_N\right) \quad , \tag{14.5}$$

where x_N is the penetration distance in the normal direction, C_N is the normal stiffness coefficient, e_N is the normal restitution coefficient, v_N^- is the incidence velocity in normal direction, and $\dot{x}_N$ is the normal relative velocity. The advantage of this model is the ability to avoid discontinuity in the force function at the moment when the impact starts, that is, when velocity is pointing in the normal direction.

In the tangential direction the force can be calculated for two distinct situations [8] using Coulomb's law for friction:

- Sticking force:

$$F_{T,sticking} = C_T\,|x_T|^n \left(\frac{x_T}{|x_T|} + \frac{3}{2}\frac{1}{|\dot{x}_T^-|}\left(\frac{1}{e_T} - 1\right)\dot{x}_T\right) \tag{14.6}$$

- Sliding force:

$$F_{T,sliding} = \mu F_N \left(-\frac{\dot{x}_T}{|\dot{x}_T|}\right) \tag{14.7}$$

Here, μ is the kinetic friction, and the notation for the other symbols are similar to those in the normal case.

An exponential torsional spring-damper element is attached to the joint to model viscoelasticity between the hindfoot and forefoot:

$$T_e = K\phi^{\kappa} + d\omega \quad , \tag{14.8}$$

where T_e is the applied torque, K the stiffness coefficient, ϕ the joint angle, κ the exponential coefficient of the spring, and d the damping coefficient.

The other spring-damper element features a piecewise linear moment/rotation behavior:

$$T_l = \begin{cases} -T_0 & \phi \leq -\phi_0 \\ \frac{\phi}{-\phi_0}(-T_0) & -\phi_0 < \phi \leq 0 \\ 0 & 0 < \phi \end{cases} , \qquad (14.9)$$

where T_l is again the applied torque, and T_0 and ϕ_0 are constant positive coefficients for the torque and the joint angle. Defined in this way, the element has a soft behavior for plantarflexion (negative) rotation and a stiff behavior for dorsiflexion (positive) rotation of the forefoot (cf. Figure 14.3).

The foot-ground contact can be modeled as cylinder-plane contact [13]. This method provides the advantages of a smooth and simple contact dynamics. Three cases can be identified (see Figure 14.4):

- The edge of the cylinder (foot segment) is in contact with the plane (ground): in this case the contact point lies on the edge of the cylinder
- The cylinder is almost parallel to the plane: the contact point lies between the center axis and the edge within the front face of the cylinder P.
- The cylinder is parallel to the plane: here it is assumed that the contact point is exactly the center M of the front side of the cylinder.

In order for the transition between these states to be smooth, an exponential blending function is used to smoothly interpolate the position of the contact point:

$$r = r_0(1 - e^{-C\sin\alpha}) \qquad (14.10)$$

where r_0 is the distance between M and P, α is the angle between the axis of the cylinder and the normal of the ground plane, and C is a constant that can be adjusted by the user.

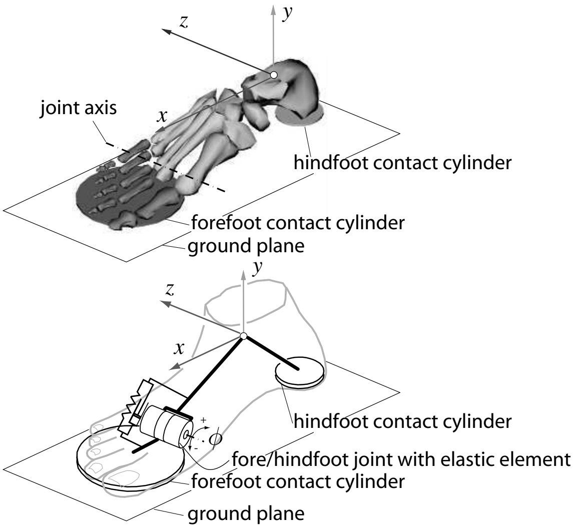

z
y
joint axis
x
hindfoot contact cylinder
forefoot contact cylinder
ground plane
y
z
x
hindfoot contact cylinder
+
φ
-
fore/hindfoot joint with elastic element
forefoot contact cylinder
ground plane

Fig. 14.3: The proposed foot model

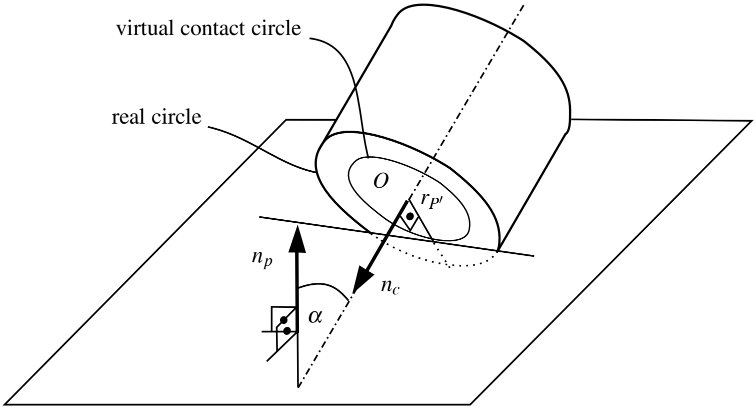


Fig. 14.4: Cylinder-plane contact model

The foot model has a large number of parameters including the joint stiffness (normal and tangential), restitution coefficients (normal and tangential), exponential coefficient for the exponential springer/damper, and the radii of the surrogate cylinders for the forefoot and the hindfoot. These parameters were determined using a preprocessing step using least-squares method to minimize the error between the measured and simulated data.

14.3.2 Simplified Experimental Settings

The stability of the standing using the two cylinders foot model was investigated using the biomechanics modeling and analysis software *MobileBody* [25]. For this purpose, the lower extremities starting from the pelvis to the feet were incorporated (cf. Figure 14.2). The hip, knee, and ankle joints were modeled as ideal spherical joints. A sinusoidal force was exerted on the pelvis in the anterior-posterior direction (normal to the frontal plane) to work as perturbation, at the same time a PID controller stabilized the model by considering each bone in the lower extremity as an inverted pendulum, and correcting the flexion-extension and the abduction-adduction angles at the hip, knee, and ankle joints (for simplicity, the rotations about the bone longitudinal axes were ignored). At the feet, the two-cylinder model stabilizes itself by reaching a static equilibrium state. This setting was designed under the assumption that the dynamics of lower extremities can be approximated by an inverted pendulum which has been shown to be valid [7]. The parameters of the PID controller were determined using the pole placement method [3]. The results are shown in [16], [25] and are not repeated here.

14.4 Characterizing Uncertainties for the Problem of Stance Stabilization

In this section, we describe how the uncertainties in several measured parameters influence the model for the human stance stabilization. At each stage of the process, there are characteristics known with some large or small incertitude (cf. Table 14.2). For example, the pelvis mass of [35,65] kg or its position on the x axis ([0.05,0.1] m) constitute the group of mass parameters and belong roughly to the first stage of the human stance modeling (cf. Section 14.3). Such parameters as the radii of forefoot or hindfoot or static and dynamic friction coefficients influence mainly the second stage. Forces along the x and y axes can be counted to the last stage. This problem has 26 degrees of freedom.

We touch upon the implementation aspects for the original model and then describe how parameter uncertainties influence the equations of motion in the first integration interval. We analyze the model in the settings from Section 14.3.2.

Table 14.2: Some uncertain parameters in stance stabilization.

Force related parameters		
ω	[0.5, 6.28]s^{-1}	frequency
F_x	[0, 200] N	force along the x axis
F_y	[0, 50] N	force along the y axis

Mass related parameters		
m_p	[35, 65] kg	pelvis mass
p_x	[0.05, 0.1] m	x-position of pelvis
p_y	[0.1, 0.5] m	y-position
p_z	[-0.05, 0.05] m	z-position

Contact related parameters		
r_{ff}	[0.04, 0.2] m	radius of forefoot
r_{hf}	[0.02, 0.15] m	radius of hindfoot
e_N	[0.01, 0.2]	normal restitution
e_T	[0.01, 0.2]	tangential restitution
μ_{st}	[0.5, 2.0]	static friction coefficient
μ_d	[0.08, 2.3]	dynamic friction coefficient

14.4.1 Implementation Issues

We implemented the model described above in SmartMOBILE. The necessary transmission elements and their parameters are imported into a C++ executable model from an XML file using the XERCES-C++ XML parser [21]. To be able to read interval-related data from the XML description directly, we extended the original XML tags with additional attributes `deviation`, `variableNR` and `variableAll` (cf. Fig. 14.5). In this Figure, the description of the transmission element `TMbRigidBodyPart` for modeling the pelvis is shown. Now it is possible to specify value ranges for the mass of the pelvis and its position with the help of the attribute `deviation` in the tags `<mass>`, `<x>`, `<y>` and `<z>`. For example, the mass is equal to 50 ± 15 kg, that is, it lies in the interval $[35,65]$kg. Additionally, we fix the mass of the pelvis as the only parameter with respect to which we would like to compute sensitivities by setting the attribute variableNR (the current variable number) to zero and the attribute variableAll (the overall number of variables) to one. We used the basic data type Finterval derived from the FADBAD++ data type F<INTERVAL> for the interval based sensitivity analysis.

```
<RigidBodyPart>
<name>Pelvis</name>
<mass deviation=”15” variableNr=”0” variableAll=”1”>50</mass><!-->[35,65]</-->
<positionOfCenterOfMass>
   <x deviation=”0.075”>0.025</x><!-->[-0.05,0.10]</-->
   <y deviation=”0.200”>0.300</y><!-->[ 0.10,0.50]</-->
   <z deviation=”0.050”>0.000</z><!-->[-0.05,0.05]</-->
</positionOfCenterOfMass>
</RigidBodyPart>
```

Fig. 14.5: The extended XML tag for the transmission element TMbRigidBodyPart (abridged).

Besides TMbRigidbodyPart for modeling bones which are connected by cartilage only and can be regarded as rigid (in a first approximation), the most important transmission elements used in the modeling process for the problem of human stance stabilization are the following:

- TMbHipJoint, TMbSphericalKneeJoint, TMbSphericalAnkleJoint for representing the full hip, knee or ankle joint motion, respectively, using the three elementary rotations flexion/extension, adduction/abduction and medial/lateral rotation;
- TMb2CylinderFootContact for modeling the foot contact as described in Section 14.4, depending among other elements on TMoRegImpCirclePlane for describing the cylinder/plane contact and the two spring dampers TMoExponentialSpringDamper (cf. Eq. (14.8)) and TMoTableSpringDamperND (cf. Eq. (14.9));
- TMbPIDWrenchController representing the necessary proportional-integral-derivative controllers.

We need the class pwFunc to represent piecewise functions such as $|x|$ or $\mathrm{sign}(x)$ and the function from the Eq. (14.9) for the TMoTableSpringDamperND. We chose $T_0 = 1000$ N·m and $\phi_0 = 0.1$ rad for the latter element. For example, if ϕ is in the range of $[-0.25, -0.05]$ rad, the function doForce() of the spring damper returns the interval force value $[-1000, -500]$, which is a true range for the function (14.9) and in accordance with the definition (14.3). However, the first derivative of (14.9) with respect to $\phi = [-0.25, -0.05]$ is equal to the interval $[0, 10000]$, which again corresponds to the definition (14.4), but overestimates the actual range $[0,0] \cup [10000, 10000]$ because the current implementation of pwFunc interprets the union as the convex hull.

Table 14.3: The abridged force vector $[w_1\ w_2\ w_4\ w_6]$ for different sets of uncertain parameters (directed rounding to the second digit after the decimal point).

	all parameters from Tab. 14.2 uncertain	m_p, p_x and F_x uncertain	nominal
w_1	[0,200] N·m	[0,200]N·m	[99.99,100.00]N·m
w_2	[-940.00,-595.69]N·m	[-915.00,-620.69]N·m	[-767.85,-767.84]N·m
w_4	[-31.89,31.89]N	[0,0]N	[0,0]N
w_6	[-50.17,45.49]N	[-50.17,45.49]N	[1.33,1.34]N

14.4.2 Influence of Uncertain Parameters on Equations of Motion

The goal was to obtain the equations of motion for the problem of stance stabilization at the first simulation time-interval to study the influence of the uncertainty in parameters on them.

The parameters of interest are the pelvis mass m_p, the position of the pelvis center of mass on the x axis p_x, the applied force along the x axis F_x and the mass of the right femur $m_{rf} = 10.34$kg. We consider the first, second, fourth and sixth coordinates w_1, w_2, w_4 and w_6 of the wrench vector from the equations of motion (moments about the x and y axes, forces along the x and z axes, respectively). In Table 14.3, we show interval evaluations for these characteristics under influence of two sets of uncertain parameters and for nominal parameters. The term *nominal parameters* means that midpoints of respective parameter ranges, represented as point intervals, were considered in computations. The sensitivity of w_1, w_2, w_4 and w_6 to m_p, p_x, F_x and m_{rf} under uncertainty in m_p, p_x and F_x is shown in Table 14.4. As a comparison, we computed the sensitivities for nominal parameters along with the resulting reference uncertainty (cf. Table 14.5).

Table 14.4: Interval sensitivity (directed rounding to the second digit after the decimal point).

	$\partial(\cdot)/\partial m_p$	$\partial(\cdot)/\partial p_x$	$\partial(\cdot)/\partial m_{rf}$	$\partial(\cdot)/\partial F_x$
w_1	0.0 N·m·kg^{-1}	0.0 N	0.0 N·m·kg^{-1}	[0.99,1] m
w_2	[-9.81,-9.80] N·m·kg^{-1}	0.0 N	[-9.81,-9.80] N·m·kg^{-1}	0.0 m
w_4	0.0 N·kg^{-1}	0.0 N·m^{-1}	[0.78, 0.79] N·kg^{-1}	0.0
w_6	[-9.82,0.50] N·kg^{-1}	[-637.66,-343.34] N·m^{-1}	[0.49,0.5] N·kg^{-1}	0.0

Table 14.5: Reference uncertainty (directed rounding to the second digit after the decimal point).

	$\partial(\cdot)/\partial m_p$	$\partial(\cdot)/\partial p_x$	$\partial(\cdot)/\partial m_{rf}$	$\partial(\cdot)/\partial F_x$	[r]
w_1	0.0 N·m·kg^{-1}	0.0 N	0.0 N·m·kg^{-1}	1.0 m	[0.00,200.00] N· m
w_2	-9.81 N·m·kg^{-1}	0.0 N	-9.81 N·m·kg^{-1}	0.0 m	[444.43,738.44] N· m
w_4	0.0 N·kg^{-1}	0.0 N·m^{-1}	0.78 N·kg^{-1}	0.0	8.07 N
w_6	-0.25 N·kg^{-1}	-490.5 N·m^{-1}	0.5 N·kg^{-1}	0.0	[38.44,70.47] N

The tables show that the force-induced part of equations of motion depends most substantially on the position and the mass of the pelvis. This holds especially for w_6 (force in z direction), which is most sensitive to p_x.

We can confirm this fact by evaluating w_6 for two uncertain parameters m_p and p_x separately. If we consider only the uncertainty in m_p, the width of the input uncertainty equals 15 kg, which is considerable. However, w_6 is bounded by the interval $[-2.35, 5.02]$ N of the acceptable width 7.37 N. As a comparison, the input uncertainty of the width 0.05 m in p_x leads to the output $[-35.46, 38.15]$ N of the diameter 73.61 N, which is not within biomechanical general tolerances. Note that this behavior corresponds to the values of sensitivities in Tab. 14.4: The partial derivative $|\partial w_6/\partial m_p|$ for nominal parameters is equal to 0.25, whereas the corresponding value for $\partial w_6/\partial p_x$ equals 490.5.

Finally, it should be mentioned that variations in ϕ from Eq. (14.9) induce a considerable overestimation in w_6 (along with w_7,w_9 and w_{11}) in the interval case. If we force the joint angle ϕ for the right foot to be in the range of $[-0.25, -0.05]$ rad in the first time-interval, the corresponding value for w_6 is $[-1638, 1641]$ N. As a comparison, the outcome for w_1 under the same conditions is $[99.16, 100.84]$ N·m. However, the corresponding sensitivity values in the nominal case are of the same magnitude, $|\partial w_1/\partial \phi| = [-10^{-12}, 10^{-12}]$ and $|\partial w_6/\partial \phi| = -0.04$ for $\phi = -0.25$ rad as well as $|\partial w_1/\partial \phi| = [-10^{-12}, 10^{-12}]$ $|\partial w_6/\partial \phi| = 0.03$ for $\phi = -0.05$ rad. This is an example of difficulties appearing while working with piecewise functions in the interval case: The true force range $[-1000, -500]$ for the spring damper element described in Eq. (14.9) is too wide for computations that follow. Note that this problem is independent of the current implementation of the class `pwFunc`. As a solution, we might want to use a different basic data type in the future.

The results show that contact mechanics can lead to large variations of dependent data when particular parameters are varied. This behavior might be damped by numerical simulation, leading to slightly smaller variations in the integrated dynamical behavior. We plan to perform this analysis in subsequent steps based on the results described in the paper.

14.5 Conclusions and Outlook

We presented a first verified sensitivity analysis of the stance stabilization model from PROREOP. We used SMARTMOBILE for this purpose, a tool providing verified kinematics, dynamics and sensitivity analysis options for several classes of (bio)mechanical systems. Besides, we introduced an implementation `pwFunc` of a class for computing interval evaluations and first derivatives of piecewise functions. We showed that the equations of motion for the stance stabilization are particularly sensitive to the position of pelvis and the pelvis mass.

The major challenge while simulating dynamics of the stance stabilization in a verified way is the foot contact stage. The main reason is that the equations of motion for it change their right side as well as in some cases their left side in dependence on the zeros of a certain switching function. For this reason, one needs to handle a hybrid system in which one switches between different modes of operation. Verified treatment of such situations is infrequent, but have some advantages, for example, for contact area modeling. For example, the contact between a cylinder and a plane is not a point but a small area for small angles between the corresponding normals, whose center of area can be again projected into a point (cf. Figure 14.4). Verified methods could offer a possibility to work with the original contact area as an interval without the need of projecting it to a point.

Our future work will include the development of a verified solver for hybrid systems, modeling of the contact area between a cylinder and a plane with the help of intervals and refinement of our implementation of the `pwFunc` class by working with disjoint intervals.

Acknowledgements This work was funded by the Deutsche Forschungsgemeinschaft.

References

1. Auer, E., Luther, W.: SmartMOBILE and its Applications to Guaranteed Modeling and Simulation of Mechanical Systems. In: Lecture Notes in Electrical Engineering, vol. 24. Springer (2009)
2. Bell, B.M.: http://www.coin-or.org/CppAD/. Web page (2006)
3. Cominos, P., Munro, N.: PID controllers: recent tuning methods and design to specification. Control Theory and Applications, IEE Proceedings - **149**(1), 46 –53 (2002). DOI 10.1049/ip-cta:20020103
4. De Momi, E., Pavan, E., Motyl, B., Bandera, C., Frigo, C.: Hip joint anatomy virtual and stereolithographic reconstruction for preoperative planning of total hip replacement. In: International Congress Series, vol. 1281, pp. 708–712. Elsevier (2005)
5. Eble, I.: Über Taylor-Modelle. Ph.D. thesis, Universität Karlsruhe (2007). In German
6. de Figueiredo, L., Stolfi, J.: Self-Validated Numerical Methods and Applications. IMPA, Rio de Janeiro (1997)
7. Gage, W.H., Winter, D.A., Frank, J.S., Adkin, A.L.: Kinematic and kinetic validity of the inverted pendulum model in quiet standing. Gait & Posture **19**(2), 124 – 132 (2004)
8. Gilardi, G., Sharf, I.: Literature survey of contact dynamics modelling. Mechanism and Machine Theory **37**(10), 1213 – 1239 (2002). DOI DOI:10.1016/S0094-114X(02)00045-9. URL http://www.sciencedirect.com/science/article/B6V46-462BFCD-4/2/578c0bad5b5f9ef5781b71a8a6896834
9. Gilchrist, L.A., Winter, D.A.: A two-part, viscoelastic foot model for use in gait simulations. Journal of Biomechanics **29**(6), 795 – 798 (1996)
10. Griewank, A.: Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation. SIAM (2000)
11. Jenkyn, T., Nicol, A.: A multi-segment kinematic model of the foot with a novel definition of forefoot motion for use in clinical gait analysis during walking. Journal of Biomechanics **40**(14), 3271 – 3278 (2007)
12. Kecskeméthy, A.: Objektorientierte Modellierung der Dynamik von Mehrkörpersystemen mit Hilfe von Übertragungselementen. Ph.D. thesis, Gerhard Mercator Universität Duisburg (1993). In German
13. Kecskeméthy, A., Lange, C., Grabner, G.: A geometric model for cylinder-cylinder impact with application to vertebrae motion simulation. In: 7th International Symposium on Advances in Robot Kinematics (2000)

14. Knüppel, O.: PROFIL/BIAS — A Fast Interval Library. Computing **53**, 277–287 (1994)
15. Lattanzi, R., Viceconti, M., Zannoni, C., Quadrani, P., Toni, A.: Hip-Op: an innovative software to plan total hip replacement surgery. Medical informatics and the internet in medicine **27**(2), 71–83 (2002)
16. Liu, X., Kecskeméthy, A., Tändl, M.: A self-stablilized foot-ground contact model using two segments and cylinder-plane pairs. (2008). I-FAB Poster
17. Makino, K., Berz, M.: Taylor models and other validated functional inclusion methods. International Journal of Pure and Applied Mathematics **4**(4), 379–456 (2003)
18. MobileBody: Patientenindividuelle Ganganalyse. http://www.uni-due.de/mechanikb/forschung/projekte.php (2010)
19. Moore, E., Kearfott, B., Cloud, M.: Introduction to Interval Analysis, vol. 1. Society for Industrial Mathematics (2009)
20. Nedialkov, N.S.: The design and implementation of an object-oriented validated ODE solver. Kluwer Academic Publishers (2002)
21. Project, T.A.X.: http://xerces.apache.org/xerces-c/. Web page
22. Rauh, A., Auer, E., Hofer, E.P.: VALENCIA-IVP: A Comparison with Other Initial Value Problem Solvers. In: Proceedings of SCAN 2006 (2007)
23. Scott, S.H., Winter, D.A.: Biomechanical model of the human foot: kinematics and kinetics during the stance phase of walking. Journal of Biomechanics **26**(9), 1091 – 1104 (1993)
24. Stauning, O., Bendtsen, C.: Fadbad++ web page. http://www.fadbad.com/
25. Tändl, M., Stark, T., Erol, N., Löer, F., Kecskeméthy, A.: An object-oriented approach for simulation of human gait motion based on motion tracking. International Journal of Applied Mathematics and Computer Science (AMCS) **19**(3), 469–483 (2009)

Chapter 15
Adaptive Control Strategies in Heat Transfer Problems with Parameter Uncertainties Based on a Projective Approach

Vasily V. Saurin (✉), Georgy V. Kostin, Andreas Rauh, and Harald Aschemann

Abstract Control problems for distributed heating systems described by parabolic partial differential equations are considered in this paper. This type of mathematical model is also a common description for other distributed parameter systems involving diffusion as well as heat and mass transfer. The goal of the paper is to develop an adaptive strategy including online parameter identification for efficient control of heat transfer systems. The developed strategy is based on the method of integrodifferential relations, a projective approach, and a suitable finite element technique. An adaptive control algorithm with predictive estimates of the desired output trajectories is proposed and its specific features are discussed. We use the parameters, geometry, and actuation principles of a real test setup available at the University of Rostock for the numerical simulation and verification. The test setup consists of a metallic rod equipped with a finite number of Peltier elements which are used as distributed control inputs allowing for active cooling and heating. A validation of the control laws derived in this contribution is performed taking into account the explicit local and integral error estimates resulting directly from the method of integrodifferential relations.

Vasily V. Saurin
Laboratory of Mechanics and Optimization of Structures, Institute for Problems in Mechanics of the Russian Academy of Sciences, Pr. Vernadskogo 101-1, 119526 Moscow, Russia
e-mail: saurin@ipmnet.ru

Georgy V. Kostin
Laboratory of Mechanics of Controlled Systems, Institute for Problems in Mechanics of the Russian Academy of Sciences, Pr. Vernadskogo 101-1, 119526 Moscow, Russia
e-mail: kostin@ipmnet.ru

Andreas Rauh
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Andreas.Rauh@uni-rostock.de

Harald Aschemann
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Harald.Aschemann@uni-rostock.de

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_15,

15.1 Introduction

The design of control strategies for dynamic systems with distributed parameters has been actively studied in recent years. Processes such as oscillations, heat transfer, diffusion, and convection are part of a large variety of applications in science and engineering. The theoretical foundation for optimal control problems with linear partial differential equations (PDEs) and convex functionals was established by Lions [1–4]. Linear hyperbolic equations are treated, besides in Lions' book, in [5,6]. An introduction to the control of vibrations can be found in [7]. Oscillating elastic networks are investigated in [8–10].

Since accurate modeling of these systems leads to a description in terms of PDEs, control design is usually based on specific approaches to solving direct and inverse problems. In most cases, it is necessary to develop advanced control strategies.

15.1.1 Variational Formulations and Projective Approaches

In cases where the PDEs and space domains are linear and simple, it is possible to obtain the solution in so-called closed form, often as an infinite series in which the terms are given by a product of one function only depending on time and others depending on the spatial coordinates. The conventional approaches for the analysis of such systems are known as the Fourier and Laplace methods [11]. For solving more complicated problems (e.g., systems with non-homogeneous and nonlinear properties and irregular shapes), variational and projective approaches have been thoroughly developed and studied by scientists. It is rather usual that the PDEs describing different physical phenomena are stationary conditions of some variational problem. Among these formulations, the Hamilton principle in dynamics corresponding to the minimum principle for potential energy of static problems can be mentioned (cf. [12]). Alternative variational principles for initial-boundary value problems were obtained on the basis of the Laplace transformation in [13].

Another approach is the method of integrodifferential relations (MIDR) which has been presented in [14] and applied to static problems regarding the linear theory of elasticity. One of the important features of the MIDR is that an original boundary or initial-boundary value problem in PDEs can be reduced to a variational problem: minimize a non-negative functional following from constitutive relations such as Hooke's law in the case of linear elasticity or Fourier's law for heat transfer. During evaluation of the system model, the value of the functional can serve as an integral estimate for the quality of any approximate solution, whereas the integrand characterizes the local error distribution.

Variational formulations in finite element methods are frequently used in scientific and engineering applications. The mathematical origin of this method can be traced back to a paper by Courant [15]. Other numerical approaches, e.g., the Petrov-Galerkin method [16,17] or the least squares method [18], are being developed actively for numerical modeling of dynamical processes. Various a priori and a posteriori heuristic criteria have been applied to improve the solution quality [19].

The numerical algorithm based on the MIDR and variational techniques was worked out and applied to linear elasticity problems [20–22]. The FEM realization gives one the possibility to develop various strategies for adaptive mesh refinement by using a local error estimate [23]. Separated approximations including on the one hand an expansion of finite dimension over some coordinate components and on the other hand unknown functions over one remaining component can be used in the MIDR [24]. For this expansion, the original problem described by PDEs is reduced to a finite-dimensional system of ordinary differential equations (ODEs). This approach has been applied to heat conduction problems [25], where the first law of thermodynamics and the corresponding initial and boundary conditions are exactly satisfied, while Fourier's law is given in a weak form.

Various projective approaches which are based on the MIDR can also be developed and applied effectively to reliable numerical modeling of physical processes. In contrast to the variational technique, projections of constitutive relations on a functional space that is chosen in a special way are used to compose a consistent system of equations. A modification of the MIDR which is based on this projective technique and an ansatz representation of unknown functions is derived in [26]. A corresponding numerical algorithm has been developed to define the temperature profile and heat flux density for one-dimensional heat transfer problems.

15.1.2 Early vs. Late Lumping

There are basically two different approaches to the control design for distributed parameter processes.

In the first approach, often called *late lumping*, the control laws are directly designed for the distributed parameter models and then converted to a finite approximation. Note that the infinite-dimensional control strategies rely on specific spectrum analysis of the linear system operator [27,28]. The control method considered in [29,30] enables one to construct a constrained distributed control in closed form and ensures that the system is brought to a given state in a finite time. This method is based on a decomposition of the original system into simple subsystems by the Fourier approach. In [31], a numerical approach for the solution of PDE-constrained optimal control problems is adapted to hyperbolic equations. The method of choice proposed there is either a full discretization method, in case of small size problems, or the vertical method of lines, in case of medium size problems.

In applications, the second approach, so-called *early lumping*, is broadly used for numerical control design if the mathematical models are given in the form of PDEs. In this way, the initial-boundary value problem is first discretized and reduced to a system of ODEs by means of the Rayleigh-Ritz or the Galerkin methods. Alternatively, finite difference or finite element schemes as well as other model approximation techniques can be used as shown in [32,33]. The direct discretization methods are also well known in control problems (see, e.g. [34]). A family of Galerkin approximations based on solutions of the homogeneous beam equation was constructed and sufficient conditions for stabilizability of such finite-dimensional systems were derived in [35]. In addition, the equilibrium of the Galerkin approximation considered is proven to be stabilizable by an observer-based feedback control law and an explicit control design is proposed.

The design of control strategies for the distributed heating system which is in the focus of this paper has already been studied in previous publications on the basis of the early lumping ideas. For example, in [36,37], a procedure for the numerical computation of a different feedforward control strategy has been presented which makes use of a finite-volume discretization of the heat transfer equation in order to replace the parabolic PDE by a set of ODEs. For this spatially discretized system model, both classical numeric and novel interval arithmetic solvers for sets of differential algebraic equations (DAEs) have been implemented to compute desired trajectories and control inputs in such a way that the output temperature of the system at a specific position matches a predefined time response. Moreover, the interval-based DAE solver VALENCIA-IVP has been used in [36, 37] to verify the estimation quality of classical estimator concepts which in the presence of bounded measurement noise can be employed for online identification of internal system states which are not measured directly or which are not directly accessible for measurements. The algorithmic details of VALENCIA-IVP have been presented in [36]. Interval tools for verified sensitivity analysis as well as verified reachability analysis and observability analysis are summarized in [38].

One of the drawbacks of the early lumping approach is that it is rather difficult to know the connection between the original distributed parameter model and its discretized version a priori. However, this connection can be qualified by the explicit error estimates following directly from the MIDR formulation for inverse problems as shown in [25, 26, 39]. These estimates allow one to verify the quality of the finite-dimensional modeling, to refine numerical solutions and to make corresponding corrections of the control laws online.

15.1.3 Advanced Control Strategies

Among various control strategies, it is worth to note two basic directions, namely, feedforward and feedback. Feedforward control can eliminate, in the ideal situation, or at least reduce the effect of measured disturbances on the dynamical process if accurate system models are known and their initial states are consistent with the desired trajectories. A feedback control system is required to suppress undesirable measured as well as unmeasured disturbances that are always present in any real process. The combination of feedforward and feedback control can significantly improve system performance.

Powerful approaches to system analysis, trajectory planning, and feedforward as well as feedback control have been derived after extending the method of flatness-based control from finite to infinite-dimensional systems. The combination of backstepping-based state-feedback control and flatness-based trajectory planning and feedforward control is considered in [40] for the design of an exponentially stabilizing tracking controller for a linear diffusion-convection-reaction system with parameters varying in space and time and a nonlinear boundary input. In [41], two new flatness-based tracking control strategies are developed, which are numerically efficient and applicable online. In these approaches, the PDE is transformed into an ODE for every measuring point using the method of characteristics. The flatness-based solution procedures proposed in [42–44] for systems with distributed and boundary control inputs are based on a mathematical discretization of the PDE by an ansatz function separating the dependencies on time and spatial coordinates.

Adaptive control is becoming popular in many fields of engineering and science and faces many important challenges, especially in real-time applications for distributed parameter systems, which do not have precise models applicable to control design. In [45], some common and efficient adaptive control approaches, including model reference adaptive control, adaptive pole placement control, and adaptive backstepping control are presented and analyzed.

The book [46] introduces a comprehensive methodology for adaptive control design of parabolic PDEs with unknown functional parameters, including reaction-convection-diffusion systems ubiquitous in chemical, thermal, biomedical, aerospace, and energy systems.

In Section 15.2, the statement of an initial-boundary value problem for parabolic PDEs is given. The projective approach based on the MIDR is discussed in Section 15.3. After that, the finite element algorithm is proposed in the frame of this approach in Section 15.4. In the next section, after formulation of the control problem for tracking of a desired temperature profile, both pure feedforward and adaptive control strategies are developed. The test setup and actual control structure are described in Section 15.6. The robustness of the adaptive control strategy proposed is demonstrated and numerically verified in Section 15.7. Finally, the paper is concluded with an outlook on future research in Section 15.8.

15.2 Mathematical Statement of the Heat Transfer Problem

Consider a one-dimensional heat transfer process in a rod with length l . The heat flux law (Fourier's law) relates the heat flux density $q(z,t)$ and the temperature gradient to each other according to

$$\xi(\vartheta,q) := q + \lambda \frac{\partial \vartheta}{\partial z} = 0\,. \tag{15.1}$$

In this equation, the temperature is denoted by $\vartheta(z,t)$ and λ is the heat conductance.

The first law of thermodynamics leads to

$$\frac{\partial q}{\partial z} + \kappa_1 \frac{\partial \vartheta}{\partial t} + \kappa_2 \vartheta = \mu(z,t) , \tag{15.2}$$

where κ_1 and κ_2 are some physical coefficients characterizing the heat capacity and heat transfer, respectively. The function $\mu(z,t)$ represents both the distributed control and external disturbances.

In terms of the heat flux density q , the boundary conditions are given by

$$q(0,t) = \bar{q}_0(t) \quad \text{and} \quad q(l,t) = \bar{q}_l(t) . \tag{15.3}$$

In Eq. 15.3, $\bar{q}_0$ and $\bar{q}_l$ are given functions. For example, if $\bar{q}_0(t) = 0$ holds, adiabatic insulation of the rod end at the position $z = 0$ is taken into account.

To close the formulation of an initial-boundary value problem, let us specify the initial temperature distribution by

$$\vartheta(z,0) = \bar{\vartheta}_0(z) . \tag{15.4}$$

Integrating Eq. 15.2 with respect to the coordinate z and taking into account the first boundary condition in Eq. 15.3 leads to the explicit expression for the heat flux density

$$q(z,t) = \int_0^z \left[\mu(x,t) - \kappa_1 \frac{\partial \vartheta}{\partial t} - \kappa_2 \vartheta \right] dx + \bar{q}_0(t) . \tag{15.5}$$

Then, the second boundary condition in Eq. 15.3 takes the form of a linear integro-differential equation

$$\int_0^l \left[\kappa_1 \frac{\partial \vartheta}{\partial t} + \kappa_2 \vartheta \right] dx = \int_0^l \mu(x,t) dx + \bar{q}_0(t) - \bar{q}_l(t) . \tag{15.6}$$

The constitutive relation (15.1) can be rewritten using Eq. 15.5 as

$$\xi = \lambda \frac{\partial \vartheta}{\partial z} + \int_0^z \left[\mu(x,t) - \kappa_1 \frac{\partial \vartheta}{\partial t} - \kappa_2 \vartheta \right] dx + \bar{q}_0(t) = 0 . \tag{15.7}$$

15.3 A Projective Approach Based on the Method of Integrodifferential Relations

15.3.1 Integrodifferential Formulation of the Heat Transfer Problem

To solve the initial-boundary value problem (15.4), (15.6), (15.7), the MIDR is applied in which the constitutive relation (15.7) is replaced by a corresponding integral relation defined as

$$\Phi = \int_0^{t_f}\int_0^{l} \varphi\, dz dt = 0 \quad \text{with} \quad \varphi = \xi^2(z,t,\vartheta)\,. \tag{15.8}$$

In (15.8), the interval $[0,t_f]$ denotes the time horizon over which the process is considered, t_f is the terminal instant of the process.

Thus, the initial-boundary value problem of the linear theory of heat conduction can be reformulated: find the admissible temperature distribution $\vartheta^*(z,t)$ that obeys the initial condition (15.4) as well as the boundary condition (15.6) and that satisfies the integral relation (15.8).

In addition to computing an approximation to the true temperature distribution, this integrodifferential formulation gives us the possibility to estimate the solution quality. Note that the integrand φ in Eq. 15.8 is non-negative. Hence, for any admissible temperature function $\tilde{\vartheta}(z,t)$ satisfying the constraints (15.4) and (15.6), the integral $\tilde{\Phi} = \Phi(\tilde{\vartheta})$ is non-negative and reaches its absolute minimum on the solution $\vartheta^*(z,t)$ (see [25]). The value $\tilde{\Phi}$ of this functional can serve as a measure for the integral quality of the approximate solution $\tilde{\vartheta}$, whereas its integrand φ shows the distribution of the local error.

The dimensionless ratio

$$\Delta = \frac{\tilde{\Phi}}{\tilde{\Psi}} \quad \text{with} \quad \tilde{\Psi} = \int_0^{t_f}\int_0^{l} \left(\lambda \frac{\partial \tilde{\vartheta}(z,t)}{\partial z}\right)^2 dz dt \tag{15.9}$$

can be used as the relative integral error of an admissible temperature field $\tilde{\vartheta}$.

15.3.2 Projective Formulation

Instead of the variational formulation following the integrodifferential problem (15.4), (15.6), (15.8), as it was shown in [25], the projective approach proposed in [26] is used in this paper to develop an adaptive control algorithm for the heat transfer system described in Section 15.5.

Using the projective approach, a weak statement of the problem is given as follows: find the temperature function $\tilde{\vartheta}^*(z,t)$ that satisfies the constraints (15.4) and (15.6) such that

$$\int_0^l (\mathscr{L}\vartheta - f)\,\eta dz = 0, \quad \vartheta(z,t) \in \mathbb{H}^1_{(0,l)}, \quad \forall\eta(z) \in \mathbb{L}^2_{(0,l)}\,,$$

$$\mathscr{L}\vartheta := \lambda\frac{\partial\vartheta(z,t)}{\partial z} - \int_0^z \left(\kappa_1\frac{\partial\vartheta(x,t)}{\partial t} + \kappa_2\vartheta(x,t)\right)dx\,, \tag{15.10}$$

$$f(z,t) := -\int_0^z \mu(x,t)dx - \bar{q}_0(t) = 0\,,$$

hold, where η is a square integrable function on the coordinate interval $z \in (0,l)$, $\mathscr{L}$ is the linear integrodifferential operator, and $f(z,t)$ is a given function.

Note that, according to [47], the Galerkin approach using the temperature discretization with respect to the space coordinate z leads to a system of ODEs in the time variable t. If the N-dimensional subspace $\mathbb{S}^1_h$ in the Sobolev space $\mathbb{H}^1_{(0,l)}$ is given, the Galerkin principle leads to the following problem statement: find the function ϑ^*_h which belongs to the trial space $\mathbb{S}^1_h$ for each $t > 0$ and obeys the projective relation

$$\int_0^l (\mathscr{L}\vartheta_h - f)\,\eta_h dz = 0 \quad \text{with} \quad \vartheta_h(z,t) \in \mathbb{S}^1_h \quad \text{for all} \quad \eta_h(z) \in \mathbb{S}^0_h \tag{15.11}$$

as well as the initial and boundary conditions (15.4) and (15.6) for any test function $\eta_h \in \mathbb{S}^0_h \subset \mathbb{L}^2_{(0,l)}$.

15.4 Finite Element Discretization

In this section, a new numerical algorithm for the finite element discretization of linear heat transfer problems is considered. The algorithm is based on the weak formulation (15.4), (15.6), (15.11) introduced in Section 15.3 and a piecewise polynomial approximation ϑ_h of an unknown temperature function ϑ .

Let the length of the rod $z \in [0,l]$ be divided into a finite number N of interval elements

$$z \in [z_{i-1}, z_i], \quad 0 = z_0 < z_1 < \ldots < z_{N-1} < z_N = l\,,$$

where z_i , $i = 0, \ldots, N$, are the nodal coordinates.

The approximation ϑ_h of the temperature profile is defined on the set of polynomial splines

$$S^1_h = \left\{ \begin{array}{r} \vartheta_h(z,t) :\ \vartheta_h = \vartheta_i = \sum\limits_{k=0}^{M} g^k_{0i}(z) g^{M-k}_{1i}(z)\theta_{ik}(t), \\ z \in [z_{i-1}, z_i], \quad i = 1, \ldots, N \end{array} \right\}, \tag{15.12}$$

where $\theta_{ik}(t)$ are unknown time-dependent coefficients and M is the fixed degree of the complete polynomial with respect to the spatial coordinate z. For any interval $z \in [z_{i-1}, z_i]$, $i = 1, \dots, N$, the linear functions g_{0i} and g_{1i} have the following form

$$g_{0i} = \frac{z_i - z}{z_i - z_{i-1}}, \quad g_{1i} = \frac{z - z_{i-1}}{z_i - z_{i-1}} .$$

This approximation should be continuous at the inner nodes z_j, $j = 1, \dots, N-1$.

Consider two intervals with a common node z_j. Due to the special representation of the polynomials in Eq. 15.12, the relation

$$\theta_{jM}(t) = \theta_{j+1,0}(t)$$

results from the continuity conditions of the temperature field. Now, it is possible to compose the vector θ of independent unknown functions

$$\theta(t) = \left\{\theta_{10}, \theta_1', \dots, \theta_N'\right\} \quad \text{with} \quad \theta_i' = \{\theta_{i1}, \dots, \theta_{iM}\}, \quad i = 1, \dots, N . \tag{15.13}$$

The dimension of the vector $\theta(t)$ is $K = MN + 1$.

Define the finite-dimensional space $\mathbb{S}_h^0$ of test functions η_h with the following basis

$$\eta_{ij} = \begin{cases} g_{0i}^j(z) g_{1i}^{M-1-j}(z), & z \in [z_{i-1}, z_i] \\ 0, & z \notin [z_{i-1}, z_i] \end{cases}, \quad i = 1, \dots, N, \quad j = 0, \dots, M-1.$$

Then, the projective relation in Eq. 15.11 is reduced to a system of $K-1$ ODEs with respect to the vector of unknown coefficients θ according to

$$\int_0^l \Big(L\vartheta_h(z, \theta(t)) - f(z,t)\Big) \eta_{ij}(z) dz = 0, \quad i = 1, \dots, N, \quad j = 0, \dots, M-1 . \tag{15.14}$$

The boundary condition (15.6) expressed as

$$\int_0^l \left[\kappa_1 \frac{\partial \vartheta_h(x, \theta(t))}{\partial t} + \kappa_2 \vartheta_h(x, \theta(t))\right] dx = -f(l,t) - \bar{q}_l(t) \tag{15.15}$$

has to be added further to the ODE system (15.14).

Projecting the initial condition (15.4) on the trace of the functional space $\mathbb{S}_h^1$ at the initial instant $t = 0$ leads to the least squares minimization

$$\theta(0) = \theta_0, \quad \int_0^l \left[\vartheta_h(z, \theta_0) - \bar{\vartheta}_0(z)\right]^2 dz \to \min_{\theta_0} . \tag{15.16}$$

according to [26].

The ODE system (15.14) with (15.15) and the condition (15.16) define a consistent initial value problem to find the approximate temperature field $\vartheta_h(z,t)$.

15.5 Optimal Feedforward and Adaptive Control with Online Parameter Identification

15.5.1 Statement of the Control Problem

For the initial-boundary value problem defined by (15.4), (15.6), and (15.11), we restrict ourselves to the case in which the function $\mu(z,t)$ can be divided into two parts according to

$$\mu = \mu_d(z,t) + \mu_c(z,t) \quad \text{with} \quad \mu_d = a_d(z)v(t) \quad \text{and} \quad \mu_c = a_c(z)u(t)\,. \qquad (15.17)$$

Here, $v(t)$ is the function of external disturbances, $u(t)$ is the control input, $a_d(z)$ and $a_c(z)$ are known functions of the spatial coordinate.

In the open-loop control problem for the system (15.4), (15.6), (15.11), and (15.17) we assume that $z = z_d$, $0 \le z_d \le l$, denotes the output position of the system. The goal of the following control strategies is the computation of a control input $u(t)$ such that the output temperature $\vartheta(z_d,t)$ coincides with a sufficiently smooth temperature profile $y_d(t)$ according to

$$\vartheta(z_d,t) \overset{!}{=} y_d(t)\,.$$

15.5.2 Optimal Feedforward Control Strategy

Different feedforward control strategies for the heat transfer system (15.4), (15.6), (15.11), (15.17) have been developed in [26]. One of the possible ways for control design is to minimize the deviation of the output temperature $\vartheta(z_d,t)$ from the desired profile $y_d(t)$. For example, if all system parameters including the initial and boundary conditions as well as the external disturbances are supposed to be known, the optimal control problem can be formulated as follows: find the function $u^*(t) \in \mathbb{U}$ that transfers the heating system from the initial state (15.4) to a terminal state in fixed time t_f and minimizes the quadratic objective function

$$J(u) = \int_0^{t_f} \Big(\vartheta(z_d,t) - y_d(t)\Big)^2 dt \to \min_{u \in \mathbb{U}}\,, \qquad (15.18)$$

where the input u belongs to the set of admissible controls $\mathbb{U}$.

In this paper, we restrict the representation of the control input $u(t)$ in the solution of the optimal control problem (15.18) to a set of time polynomials

$$\mathbb{U} = \left\{ u : \ u = \sum_{k=0}^{N_c} u_k t^k \right\} , \tag{15.19}$$

where u_k are unknown real coefficients. All control parameters u_k are collected in a vector

$$w := \{u_0, \ldots, u_{N_c}\} .$$

This representation of the control function u allows for applying the FEM technique described in Section 15.4 to find a numerical solution to the optimal control problem (15.18). After substituting the expressions (15.17) for the function $\mu(z,t)$ in f defined by Eq. 15.11 and taking into account the polynomial control (15.19), an equivalent initial value problem for the unknown vector function θ introduced in Eq. 15.13 is formulated: find a solution $\theta^*(t,w)$ that obeys, for an arbitrary control vector w, the finite dimensional ODE system (15.14) and (15.15) under the initial constraints (15.16).

The components of the vector θ^* depend on the control parameters u_k, $k = 0, \ldots, N_c$. The unknown parameters u_k can be used to find the optimal control law by means of minimization of the functional (15.18).

The vector function θ^* defines the approximation $\vartheta_h^*(z,t,w)$ of the temperature according to (15.12). After substituting the approximate solution $\vartheta_h^*(z_d,t,w)$ for $\vartheta(z_d,t)$ in Eq. 15.18 and taking into account the polynomial representation of the control function u given in Eq. 15.19, the corresponding optimization problem (15.18) is reduced to an unconstrained minimization of a quadratic function with respect to the unknown parameters u_k . The optimal vector w^* is given as the solution of the following system of linear algebraic equations

$$\frac{\partial J_h}{\partial u_k} = 0, \quad k = 0, ..., N_c ,$$

where J_h is given by

$$J_h = \int_0^{t_f} \Big(\vartheta_h^*(z_d,t,w) - y_d(t) \Big)^2 dt .$$

The functions $\vartheta_h^*(z,t,w^*)$ describe an approximate solution of the original optimal control problem (15.18) under the constraints (15.4), (15.6), (15.11), and (15.17).

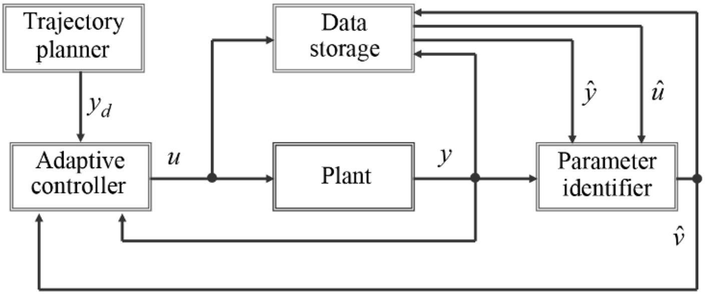


Fig. 15.1: Adaptive control structure

15.5.3 Adaptive Control Algorithm

The feedforward control algorithm described in the previous subsection has been derived under the assumption that the system parameters are known exactly. In the case of parameter uncertainties, a pure offline control strategy can lead to significant deviations of the actual output values from the desired time trajectories [26].

Various feedback control algorithms are widely used to correct the errors of feedforward control strategies. Among these approaches, adaptive algorithms that estimate the parameter uncertainty and adjust the control law online are of great importance [46]. In this paper, an adaptive control strategy taking into account the influence of unknown external disturbances is proposed. It is supposed that all the internal parameters of the heat transfer system (15.4), (15.6), (15.11), and (15.17) are given. The function of external disturbances $v(t)$ defined in Eq. 15.17 is unknown. Of course, the physical nature of such disturbances can be different and it is often impossible to predict their behavior a priori. Nevertheless, for the reason of controllability, we constrain ourselves to the case in which the function $v(t)$ changes its value slowly compared to the rate of the transient phenomena in the system.

The general scheme of the adaptive control system is depicted in Fig. 15.1. The control strategy takes into account a sequence of time steps.

At the initial time $t = 0$ the vector function of measured temperatures

$$y = \{y_1, \ldots, y_{N_y}\}, \quad y_i = \vartheta(z_i^y, t), \quad z_i^y \in [0, l], \quad i = 1, \ldots, N_y, \quad z_k^y < z_{k+1}^y \tag{15.20}$$

and a value $\tilde{v}(0)$ of the function of external disturbances are given and written in the data storage as $\hat{y}$ and $\hat{v}$, respectively.

The following control cycle is organized:

Using the current output value y, the identified external function $\tilde{v}(t)$, and the desired temperature profile $y_d(t)$ generated by the *trajectory planner*, the control $u(t)$ is found by the *adaptive controller*, written in the *data storage* as $\hat{u}$, and applied to the *plant* at the beginning of the time step $t = t_k$. At the end of this time step, the vector y is measured and used together with the saved values $\hat{y}$ and $\hat{v}$ in the

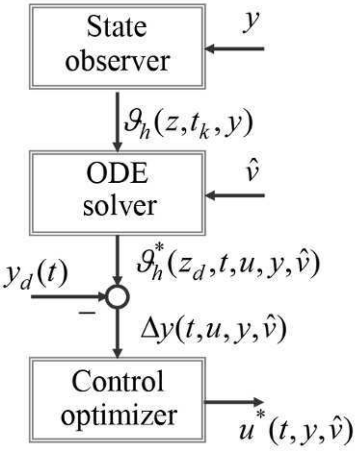


Fig. 15.2: Scheme of the adaptive controller

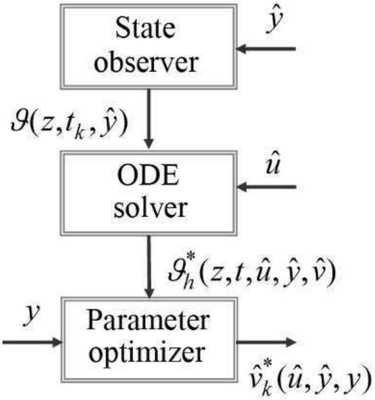

State observer
$\hat{y}$
$\vartheta(z,t_k,\hat{y})$
ODE solver
$\hat{u}$
$\vartheta^*_h(z,t,\hat{u},\hat{y},\hat{v})$
y
Parameter optimizer
$\hat{v}^*_k(\hat{u},\hat{y},y)$

Fig. 15.3: Scheme of the parameter identifier

parameter identifier to produce a new function $\tilde{v}(t)$. After that, the current vector y and the identified function $\tilde{v}(t)$ are put into the *adaptive controller*, saved in the *data storage* as $\hat{v}$, and then the control cycle is repeated in the following time step.

The principal structure of the *adaptive controller* is shown in Fig. 15.2. The vector y measured at the beginning of the current time step $t = t_k$, $k = 0,1,\ldots$, is put into the *state observer* which processes these data and generates the finite dimensional initial distribution of the temperature $\vartheta_h(z,t_k,y)$ for the *ODE solver*. In the *ODE solver*, the time history of the control output $\vartheta^*_h(z_d,t,u,y,\hat{v})$ for any admissible control u is obtained based on some approximation of the heat transfer system and the identified external function $\tilde{v}(t)$. After that, the difference $\Delta y = \vartheta^*_h(z_d,t,u,y,\hat{v}) - y_d(t)$ between the output ϑ^*_h and the desired profile values $y_d(t)$ is fed into the *control optimizer*, where the optimal control $u^* = u^*(t,y,\hat{v})$ is found.

Here, the function u^* is obtained from the following minimization problem

$$J_k(u) = \int_{t_k}^{t_k+t_p} \Delta y^2 dt \to \min_{u \in \mathbb{U}_a} , \tag{15.21}$$

where $\mathbb{U}_a$ is some control set, in particular, it can be polynomial, t_p is some predictive horizon which must guarantee the stability of the control process.

The scheme of the *parameter identifier* is presented in Fig. 15.3. The vector $\hat{y} = y(t_{k-1})$ from the *data storage* is processed by the *state observer* generating the distribution of the temperature $\vartheta_h(z,t_{k-1})$ for the *ODE solver*. Then, the *ODE solver* gives the function $\vartheta^*_h(z,t,\hat{u},\hat{y},\hat{v}_k)$ for any external disturbance $\hat{v}_k = \text{const}$ based on the system approximations and control function $\hat{u}(t)$ stored. After that, the temperature distribution at the beginning of the time step $\vartheta_h^*(z, t_k, \hat{y}, \hat{v}_k)$ and the current vector $y(t_k)$ are provided to the *parameter optimizer* which produces the identified value $\hat{v}_k^*(y, \hat{y}, \hat{u})$ of the unknown function $v(t)$. Note, that the stored values $\hat{v}_i$, $i \leq k$, can be used to refine the extrapolation $\tilde{v}(t)$. An independent choice of the control and identification time steps may also give us some additional flexibility to increase the efficiency of the control process.

The optimal parameter $\hat{v}_k^*$ is found from the minimization problem

$$J_k^y(\hat{v}_k) = \sum_{i=1}^{N_y} \left(\vartheta_h^*(z_i^y, t_k, \hat{y}, \hat{v}_k) - y_i(t_k) \right)^2 \to \min_{\hat{v}_k}, \qquad (15.22)$$

where $y_i(t_k)$ is the measured temperature at the point with coordinate z_i^y introduced in Eq. 15.20.

15.6 Test Setup and Actual Control Structure

To visualize the use of the adaptive control strategy described in the previous sections, we consider the heating system depicted in Fig. 15.4. A corresponding setup has been built up at the Chair of Mechatronics of the University of Rostock. Four Peltier elements and cooling units are distributed over the length l of an iron rod and divide it into four equally long segments. The rod temperature is measured at the midpoints $z_i^y = (2i-1)l/8$, $i = 1, \ldots, 4$, of the segments. The output point z_d is the middle of Segment 1. The insulation of the edges of the rod is assumed to be adiabatic. The manipulated variable of this heating system is the heating power u supplied with the Peltier element at Segment 4.

Mathematically, the temperature distribution is described by the one-dimensional heat transfer Eqs. 15.1, 15.2 with the parameters

$$\kappa_1 = \rho c_p \quad \text{and} \quad \kappa_2 = \frac{\alpha}{h}.$$

Here, ρ is the volume density of iron, c_p is the specific heat capacity, α is the heat transfer coefficient, and h is the height of the rod. It is supposed that the input heat flux (see Eq. 15.17) is uniformly distributed along Segment 4 and equal to zero on the other segments, thus

$$a_c(z) = \begin{cases} \dfrac{4}{bhl} & \text{for} \quad \dfrac{3l}{4} \leq z \leq l \\ 0 & \text{for} \quad 0 < z < \dfrac{3l}{4} \end{cases},$$

where b is the width of the rod. It is also considered that the ambient temperature $v(t)$ is the only external disturbance in this model and does not change its value along the rod length $a_d(z) = \kappa_2$. Due to adiabatic insulation of the edges of the rod, the equality $\bar{q}_0(t) = \bar{q}_l(t) = 0$ holds in Eq. 15.3. The initial temperature of the rod

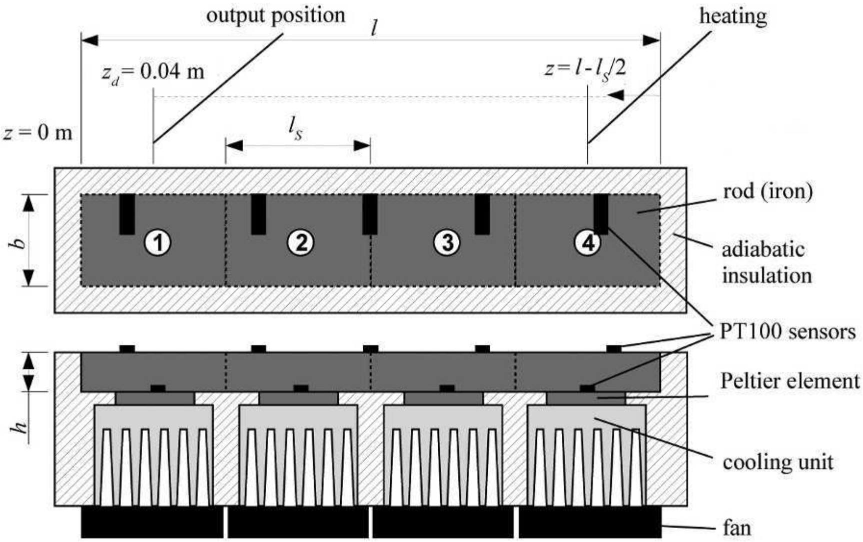


Fig. 15.4: Test setup

is distributed homogeneously ($\bar{\vartheta}_0(z) = \text{const}$) and equal to the ambient temperature $v(0) = v_0$.

The parameters of this model have been identified in experiments: $\rho = 7800\,\text{kg/m}^3$, $c_p = 420\,\text{J/(kg}\cdot\text{K)}$, $\lambda = 110\,\text{W/(m}\cdot\text{K)}$, $\alpha = 50\,\text{W/(m}^2\cdot\text{K)}$, $h = 12\,\text{mm}$, $b = 40\,\text{mm}$, $l = 320\,\text{mm}$, and $v_0 = 296.15\,\text{K}$.

In the actual adaptive control structure, the *ODE solvers* in the *adaptive controller* and the *parameter identifier* (see Fig. 15.2 and Fig. 15.3) are based on the MIDR and the FEM discretization described in Sections 15.3 and 15.4. In accordance with the experimental setup, the iron rod is divided into four finite elements ($N = 4$). The coordinates of the inner nodes are

$$z_n = \frac{nl}{4}, \quad n = 1, 2, 3\ .$$

The polynomial orders $M \geq 2$ of the temperature approximation ϑ_h in Eq. 15.12 are chosen for each element in the simulation. In each time step, the ODE system (15.14) and (15.15) is solved with the following initial condition

$$\theta(t_k) = \theta_k, \quad t_k = kt_c\ ,$$

where k is the number of the step and t_c is the control horizon.

The initial vector θ_k has to be obtained from the measurements of the temperatures at the time $t = t_k$. It can be seen from Eq. 15.13 that the dimension of θ_k is greater than the number of measurements y. So, the following system is proposed to define the vector θ_k using only four values y_i:

$$
\begin{aligned}
&1)\quad \vartheta_i(z_i^y,\theta_k) = y_i, \quad i = 1,\ldots,4\,;\\
&2)\quad \left.\frac{\partial\left(\vartheta_j(z,\theta_k) - \vartheta_{j+1}(z,\theta_k)\right)}{\partial z}\right|_{z=z_j} = 0, \quad j = 1,2,3\,;\\
&3)\quad \left.\frac{\partial \vartheta_{1,4}(z,\theta_k)}{\partial z}\right|_{z=z_{0,4}} = 0\,;\\
&4)\quad J_\vartheta = \sum_{i=1}^{4}\int\limits_{z_{i-1}}^{z_i}\left(\frac{\partial^2\vartheta_i(z,\theta_k)}{\partial z^2}\right)^2 dz \to \min_{\theta_k}\,.
\end{aligned}
\tag{15.23}
$$

Here $\vartheta_i(z_i^y,\theta_k)$, $i = 1,\ldots,4$, are the polynomial approximations of the temperature distribution defined by Eq. 15.12 for each finite element. The first condition in the system (15.23) equates the values of the approximated temperatures ϑ_i to the measured ones y_i at the midpoints of the rod segments. The second condition guarantees the smoothness of the temperature field, whereas the third one satisfies the boundary conditions (15.3) expressed via the temperature. The last relation minimizes the curvature of the temperature distribution along the rod length in an integral sense.

The piecewise constant control

$$u(t) = \hat{u}_k = \text{const}, \quad t \in [t_k, t_{k+1}]$$

is computed by the *adaptive controller*. The optimal parameter $\hat{u}_k^*$ is found from the minimization problem (15.21) in each time step. The identified function $\hat{v}(t)$

$$\hat{v}(t) = \hat{v}_k = \text{const}, \quad t \in [t_k, t_{k+1}]$$

is also considered as piecewise constant and obtained by minimization of the function J_k^y (see Eq. 15.22). The value $\hat{v}_0$ is given and equal to the initial ambient temperature v_0.

In future work, filtering approaches (such as the discrete-time Kalman filter) can be used to reduce the influence of measurement noise on the identification of the parameters $\hat{v}_t$ and θ_k. For that purpose, the estimates determined by the procedure described in this paper can serve as virtual measured data for $\hat{v}_t$ and θ_k and are then updated by a suitable filter.

Numerical simulations show that the quality of the adaptive control process is influenced significantly by the choice of the prediction horizon t_p and the control horizon t_c. It is known that the heat transfer model described by the parabolic Eqs. 15.1 and 15.2 is a system with time delay. This feature is characterized by some time parameters that can be estimated based on the analysis of the transient processes in the system. The intensity of the transients is related to the eigenvalues and eigenforms of the corresponding boundary value problem

$$\frac{\lambda}{\rho c_p}\frac{\partial^2\Theta_n}{\partial z^2}+\left(\beta_n-\frac{\alpha}{h\rho c_p}\right)\Theta_n=0\,,$$
$$\left.\frac{\partial\Theta_n}{\partial z}\right|_{z=0}=\left.\frac{\partial\Theta_n}{\partial z}\right|_{z=l}=0\,,\quad n=0,1,\dots\,.$$

The eigenvalues β_n and eigenfunctions $\Theta_n(z)$ can be found analytically according to [25] as

$$\beta_n=\frac{n^2\pi^2\lambda}{\rho c_p l^2}+\frac{\alpha}{\rho c_p h}\quad\text{and}\quad\Theta_n=\cos\left(\frac{n\pi z}{l}\right)\,.$$

The zeroth eigenvalue β_0 defines the characteristic time $\tau_0=\rho c_p h/\alpha$ of heat transfer between the rod and the environment, whereas the parameter β_1 specifies the maximal characteristic time

$$\tau_1=\frac{\rho c_p h l^2}{\pi^2\lambda h+\alpha l^2}$$

of the heat conductivity along the rod.

It is clear from a physical point of view that the transients have significant effect on the controllability of the heat transfer system, but this influence is diminishing if the distance between the input and output positions is decreasing. Let us define the characteristic relative distance for this control system as follows

$$\delta_c=\max\left\{1-\frac{z_d}{l},\left|\frac{z_4^y-z_d}{l}\right|\right\}\,.$$

Taking into account that the heat conductivity is the key physical phenomenon providing the heat transfer from the input to the output, the following estimate can be proposed for the prediction time horizon

$$t_p\geq t_p^0=\delta_c\tau_1\,.$$

Numerical computations show that using small prediction time horizons ($t_p\ll t_p^0$) leads to instability of the control process, whereas applying large time intervals ($t_p\gg t_p^0$) can increase the systematic control error induced by the ambient temperature uncertainty.

The control horizon t_c should be chosen rather small, that is

$$t_c\ll t_p\,.$$

In this case, it is possible to improve more frequently the extrapolation of the ambient temperature $\hat{v}$ and to correct the adaptive control law.

15.7 Numerical Simulation

To verify the quality of the adaptive control strategy described above, numerical simulations are performed. The resulting solutions are obtained analytically from the set of ODEs (15.14) and (15.15) under the initial conditions (15.16) using symbolic formula manipulation in MAPLE.

It is considered that the actual ambient temperature is increasing during the heating process according to the quadratic function of time

$$v(t) = v_0 + 3t^2/t_f^2\,.$$

This temperature law is used only in the numerical experiment to obtain the measured vector $y(t)$. Different extrapolations $\hat{v}(t)$ are applied for testing the control strategies described in Section 15.5.

The terminal time t_f is fixed for all considered control processes and equals $t_f = 3600$s. The desired temperature profile is given as

$$y_d\,(t) = v_0 + \frac{(\vartheta_f - v_0)}{2}\left(1 + \frac{\tanh[\sigma(t - t_f/2)]}{\tanh[\sigma t_f/2]}\right)$$

with $\vartheta_f = v_0 + 10\mathrm{K}$ and $\sigma = 0.0015$.

Three types of control strategies are investigated in the numerical simulations. The first one is the pure feedforward strategy described in Subsection 15.5.2 with the polynomial control function (15.19) and $N_c = 10$. The optimal control law $u^*(t)$ is obtained under the assumption that the ambient temperature does not change in the process, i.e., $\hat{v}(t) = v_0$.

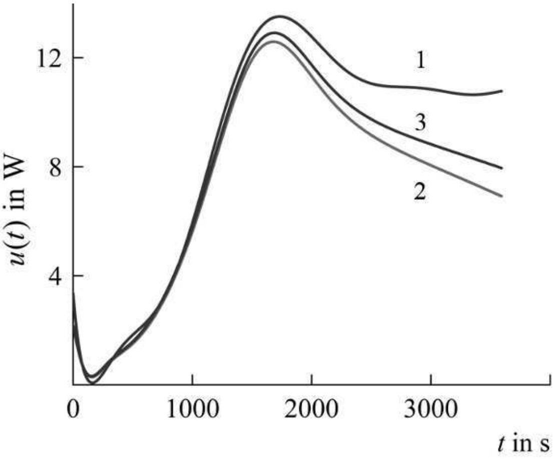


Fig. 15.5: Control functions $u(t)$: feedforward (1), adaptive with (2) and without (3) identification

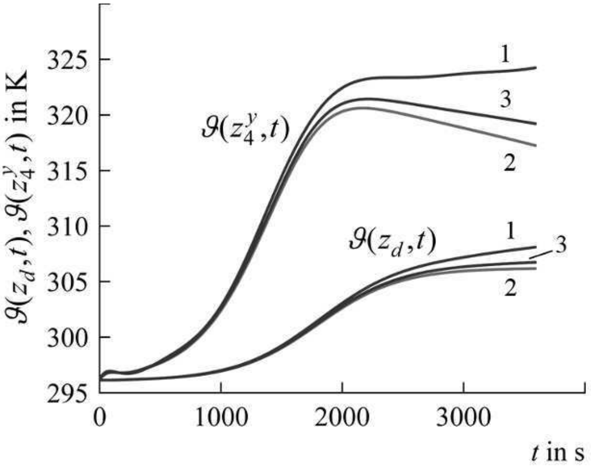


Fig. 15.6: Temperature trajectories at the input position z_4^y and the output position z_d

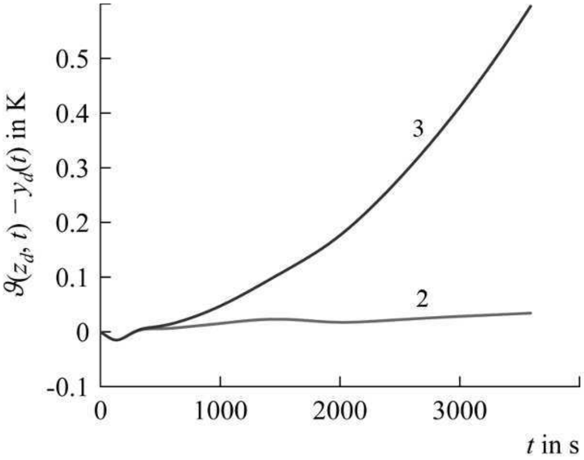


Fig. 15.7: Temperature deviations from the desired profile for the adaptive control with (2) and without identification (3)

The second and third control laws are computed during the numerical experiment based on the adaptive scheme (see Fig. 15.1) in which the *parameter identifier* is switched on and off, respectively (in the last case $\hat{v}(t) = v_0$). The prediction and control horizons are fixed to $t_p = 220\,\mathrm{s}$ and $t_c = 9\,\mathrm{s}$. Note that the characteristic time of the control system is $t_p^0 \approx 194\,\mathrm{s}$.

In Fig. 15.5, the resulting control functions (feedforward, adaptive with and without identification) are presented by the curves 1, 2, and 3, respectively.

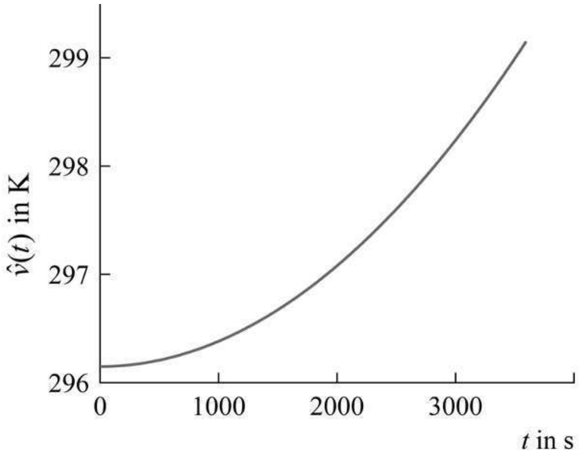


Fig. 15.8: Identified ambient temperature $\hat{v}$

The temperatures at the output position ($z = z_d$) and at the middle of the control segment ($z = z_4^y$) are shown for these control in Fig. 15.6 by curves with the same numbers.

In Fig. 15.7 the deviations of the temperature trajectories $\vartheta_1(z_d, t)$ from the desired profile $y_d(t)$ are presented for the two adaptive control strategies. The trajectory deviations for the feedforward control are rather large and not shown in this figure.

It can be seen in Figs. 15.5–15.7 that the control obtained by a pure feedforward strategy is the worst because no data about the increase of the ambient temperature is used in the optimization algorithm. In contrast, the adaptive control gives a better output trajectory even without any identification procedure, since the information about external disturbances is implicitly recognized by the *adaptive controller* on the basis of temperature measurements $y(t)$. If the adaptive strategy involves the parameter identification to estimate the ambient temperature directly, the mathematical model is corrected in the *adaptive controller* during the process and can provide more accurate output trajectories.

The identified ambient temperature $\tilde{v}(t)$ and its error $\tilde{v}(t) - v(t)$ for the adaptive strategy with identification are given in Figs. 15.8 and 15.9, respectively. The deviations of the identified temperature from its actual values (Fig. 15.9) are much smaller than the maximal change of the external temperature in the control process. The numerical simulations show that this error decreases if the control horizon t_c becomes shorter. It can also be seen that the identification accuracy goes down if the rate of the temperature growth increases. This circumstance imposes certain constraints on the applicability of the adaptive algorithm proposed.

The local error distribution $\varphi(z, t)$ introduced by the Eq. 15.8 and obtained from the numerical experiment for the adaptive control strategy with identification is depicted in Fig. 15.10 for the order $M = 2$ of the polynomial approximations on each

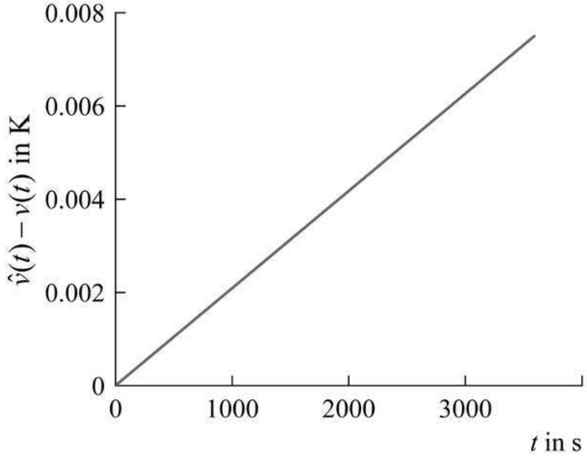


Fig. 15.9: Deviations between the identified ambient temperature and its actual values

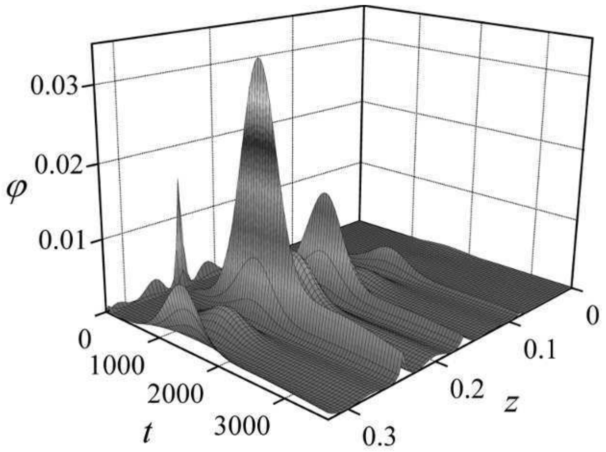


Fig. 15.10: Distribution of the local error $\varphi(z,t)$ for the temperature field $\vartheta(z,t)$

finite element. The corresponding relative integral error defined by Eq. 15.9 is sufficiently small: $\Delta = 1.5 \cdot 10^{-4}$. If the order of approximations M is increased, the integral error is decreased notably. For example, for $M = 3$ and $M = 4$ the relative errors are equal to $\Delta = 1.6 \cdot 10^{-6}$ and $\Delta = 1.1 \cdot 10^{-7}$, respectively. Note that the function $\varphi(z,t)$ helps to detect imperfection of the applied finite-dimensional model and gives one the possibility to develop new strategies for model refinement [23].

15.8 Conclusions and Outlook

In this paper, an adaptive control algorithm with parameter identification for trajectory tracking in distributed heating systems has been proposed and discussed. This control strategy is based on the method of integrodifferential relations, a projective approach, and the finite element technique. The principle scheme of the adaptive control structure has been derived and its specific features have been considered in detail. A verification of the proposed control laws is performed in numerical simulations taking into account the explicit local and integral error estimates resulting directly from the MIDR.

In future work, the basic building blocks of the control strategy proposed in this paper will be applied to more complex thermal systems. One of the goals is the use of model-based strategies for the control of the temperature distribution in the interior of high-temperature solid oxide fuel cell stacks after derivation of a suitable control-oriented model. A corresponding test rig is currently being built up at the Chair of Mechatronics of the University of Rostock. Finally, possibilities for the combination with stochastic state, parameter, and disturbance estimation will be investigated to cope with the influence of measurement noise in a more sophisticated way.

Acknowledgements This work was supported by the Russian Foundation for Basic Research, project nos. 08-01-00234, 09-01-00582, 10-01-00409 and the Leading Scientific Schools Grants NSh-3288.2010.1, NSh-64817.2010.1. This project is furthermore supported by the German Research Foundation (Deutsche Forschungsgemeinschaft DFG) under the grant number AS 132/2-1.

References

1. Lions, J.L.: Optimal Control of Systems Governed by Partial Differential Equations. Springer Verlag, New York (1971)
2. Lions, J.L.: Exact controllability, stabilization and perturbations for distributed systems. SIAM Rev. **30**(1), 1–68 (1988)
3. Tröltzsch, F.: Optimale Steuerung partieller Differentialgleichungen: Theorie, Verfahren und Anwendungen. Teubner, 2. Auflage (2010) (In German)
4. Hinze, M., Pinnau, R., Ulbrich, M.: Optimization with PDE Constraints. Springer (2009)
5. Ahmed, N.U., Teo, K.L.: Optimal Control of Distributed Parameter Systems. North Holland, New York (1981)
6. Butkovsky, A. G.: Distributed Control Systems. Elsevier, New York (1969)
7. Krabs, W.: Optimal Control of Undamped Linear Vibrations. Heldermann, Lemgo (1995)
8. Gugat, M.: Optimal control of networked hyperbolic systems: evaluation of derivatives. Adv. Model. Optim. **7**, 9–37 (2005)
9. Lagnese, J. E., Leugering, G., Schmidt, E. J. P. G.: Modeling, Analysis and Control of Dynamic Elastic Multi-Link Structures. Birkhäuser, Boston (1984)
10. Leugering, G.: A domain decomposition of optimal control problems for dynamic networks of elastic strings. Comp. Optim. Appl. **16**, 5–29 (2000)
11. Courant, R., Hilbert, D.: Methods of Mathematical Physics V. 1, Wiley (1937)
12. Washizu, K.: Variational Methods in Elasticity and Plasticity. Pergamon Press, Oxford (1982)

13. Kravchuk, A.S., Neittaanmaki, P.J.: Variational and Quasi-variational Inequalities in Mechanics. Springer, Dordrecht (2007)
14. Kostin, G.V., Saurin, V.V.: Integro-differential approach to solving problems of linear elasticity theory. Dokl. Phys. **50**(10), 535–538 (2005)
15. Courant, R.: Variational methods for the solution of problem of equilibrium and vibration. Bull. Am. Math. Soc. **49**, 1–23 (1943)
16. Atluri, S.N., Zhu, T.: A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. Comput. Mech. **22**, 117–127 (1998)
17. Belytschko, T., Lu, Y.Y., Gu, L.: Element-free Galerkin method. Int. J. Num. Methods Eng. **37**, 229-256 (1994)
18. Kwon, K.C., Park, S.H., Jiang, B.N., Youn, S.K.: The least-squares meshfree method for solving linear elastic problems. Comp. Mech. **30**, 196–211 (2003)
19. Stein, E. (ed.): Error-controlled Adaptive Finite Elements in Solid Mechanics. John Wiley, New York (2002)
20. Kostin, G.V., Saurin, V.V.: Modeling of controlled motions of an elastic rod by the method of integro-differential relations. J. Comp. Syst. Sci. Int. **45**(1), 56–63 (2006)
21. Kostin, G.V., Saurin, V.V.: The method of integrodifferential relations for linear elasticity problems. Arch. Appl. Mech. **76**(7–8), 391–402 (2006)
22. Kostin, G.V., Saurin, V.V.: Variational statement of optimization problems for elastic body motions. Dokl. Math. **76**(1), 629–633 (2007)
23. Kostin, G.V., Saurin, V.V.: A variational formulation in fracture mechanics. Int. J. Fract. **150**(1–2), 195–211 (2008).
24. Kostin, G.V., Saurin, V.V.: An asymptotic approach to the problem of the free oscillations of a beam. J. Appl. Math. Mech. **71**(4), 611–621 (2007)
25. Aschemann, H., Kostin, G.V., Rauh, A., Saurin, V.V.: Approaches to control design and optimization in heat transfer problems. J. Comp. Sys. Sci. Int. **49**(3), 380–391 (2010)
26. Rauh, A., Kostin, G.V., Aschemann, H., Saurin, V.V. Naumov, V.: Verification and experimental validation of flatness-based control for distributed heating systems. Int. Rev. Mech. Eng. **4**(2), 188–200 (2010)
27. Banks, S.P. (ed.): State-space and Frequency-domain Methods in the Control of Distributed Parameter Systems. Peregrinus, London (1983)
28. Curtain, R., Zwart, H. (eds.): An Introduction to Infinite-dimensional Linear Systems Theory. Springer Verlag, New York (1995)
29. Chernousko, F.L.: Control of elastic systems by bounded distributed forces. Appl. Math. Comp. **78**, 103–110 (1996)
30. Chernousko, F.L., Ananievski, I.M., Reshmin, S.A.: Control of Nonlinear Dynamical Systems: Methods and Applications. Springer-Verlag, Berlin-Heidelberg (1996)
31. Gerdts, M., Greif, G., Pesch, H.J.: Numerical optimal control of the wave equation: optimal boundary control of a string to rest in finite time. Math. Comput. Simul. **79**(4), 1020–1032 (2008)
32. Balas, M.J.: Finite-dimensional control of distributed parameter systems by Galerkin approximation of infinite dimensional controllers. J. of Math. Anal. and Appl. **114**, 17–36. (1986)
33. Christofides, P.D.: Nonlinear and Robust Control of PDE Systems: Methods and Applications to Transport-Reaction Processes. Birkhäuser (2001)
34. Leineweber, D., Bauer, E.I., Bock, H., Schloeder, J.: An efficient multiple shooting based reduced SQP strategy for large dynamic process optimization. Part 1: Theoretical aspects. Comp. Chem. Eng. **27**, 157–166 (2003)
35. Zuyev, A., Sawodny, O.: Stabilization and observability of a rotating Timoshenko beam model. Math. Probl. Eng., V. 2007, Article ID 31267 (2007)
36. Rauh, A., Auer, E., Aschemann, H.: Real-time application of interval methods for robust control of dynamical systems. Proc. of IEEE Intl. Conference on Methods and Models in Automation and Robotics MMAR 2009. Miedzyzdroje. Poland (2009)
37. Rauh, A., Menn, I., Aschemann, H.: Robust control with state and disturbance estimation for distributed parameter systems. In: Rodellar, J., Reithmeier, E. (eds.) Proc. of 15th Intl. Workshop on Dynamics and Control. pp. 135–142. International Center for Numerical Methods in Engineering (CIMNE), Barcelona (2009)

38. Rauh, A., Minisini, J., Hofer, E.P.: Verification techniques for sensitivity analysis and design of controllers for nonlinear dynamical systems with uncertainties. Intern. J. Appl. Math. Comp. Sci. AMCS. **19**(3) 425–439 (2009)
39. Kostin, G.V., Saurin, V.V.: The optimization of the motion of an elastic rod by the method of integro-differential relations. J. Comp. Sys. Sci. Int. **45**(2), 217–225 (2006)
40. Meurer, T., Kugi, A.: Tracking control for boundary controlled parabolic PDEs with varying parameters: Combining backstepping and differential flatness. Automatica **45**, 1182–1194 (2009)
41. Winkler, F.J., Lohmann, B.: Flatness-based control of a continuous furnace. Proc. of 3rd IEEE Multi-conference on Systems and Control (MSC), Saint Petersburg, Russia, 719-724 (2009)
42. Fliess, M., Lévine, J., Martin, P., Rouchon, P.: Flatness and defect of nonlinear systems: introductory theory and examples. Int. J. Control **61**, 1327–1361 (1995)
43. Kharitonov, A., Sawodny, O.: Flatness-based disturbance decoupling for heat and mass transfer processes with distributed control. Proc. of the IEEE International Conference on Control Applications CCA. Munich. Germany, pp. 674–679 (2006)
44. Meurer, T., Zeitz, M.: A novel design of flatness-based feedback boundary control of nonlinear reaction-diffusion systems with distributed parameters. In: Kang, W., Xiao, M., Borges, C. (eds.) New Trends in Nonlinear Dynamics and Control, Vol. 295 of Lecture Notes in Control and Information Science. Springer, 221–236 (2003)
45. Tao, G. Adaptive Control Design and Analysis. Wiley & Sons, Inc., Hoboken, New Jersey (2003)
46. Krstic, M., Smyshlyaev, A.: Adaptive Control of Parabolic PDEs. Princeton University Press (2010)
47. Strang, G., Fix, J.: An Analysis of the Finite Element Method. Prentice-Hall, Englewood (1973)

Chapter 16
State and Disturbance Estimation for Robust Control of Fast Flexible Rack Feeders

Harald Aschemann (✉), Dominik Schindele, and Jöran Ritzke

Abstract Rack feeders as automated conveying systems for high bay rackings are of high practical importance. To shorten the transport times by using trajectories with increased kinematic values accompanying control measures for a reduction of the excited structural vibrations are necessary. In this contribution, the control-oriented modeling for an experimental set-up of such a high bay rack feeder and the model-based design of a gain-scheduled feedforward and feedback control structure is presented. The rack feeder is modeled as an elastic multibody system. For the mathematical description of the bending deflections a Ritz ansatz is introduced for the first two bending modes. The tracking control design is performed separately for both axes using decentralized state space representations. Unmeasurable states as well as remaining uncertainties are estimated by a combined state and disturbance observer. Both the achievable performance and the resulting tracking accuracy of the proposed control concept are shown by measurement results from the experimental set-up.

16.1 Introduction

Rack feeders represent commonly used handling systems for the automated operation of high bay rackings. To further increase the handling capacity by shorter trans-

Harald Aschemann
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Harald.Aschemann@uni-rostock.de

Dominik Schindele
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Dominik.Schindele@uni-rostock.de

Jöran Ritzke
Chair of Mechatronics, University of Rostock, D-18059 Rostock, Germany
e-mail: Joeran.Ritzke@uni-rostock.de

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties*,
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5_16,

port times, control measures are necessary for the reduction of excited structural oscillations, see also [1]. One possible approach is given by flatness-based feedforward control, where the desired control inputs are determined by dynamic system inversion using the desired trajectories for the flat outputs as in [4] and [7]. However, both publications consider only a constant mass position in vertical direction on an elastic beam without any feedback control. A variational approach is presented in [6] to compute an optimal feedforward control for an elastic beam. Unfortunately, feedforward control alone is not sufficient to guarantee small tracking errors when model uncertainty is present or disturbances act on the system. For this reason in this contribution a gain-scheduled feedforward and feedback control design is presented for fast trajectory control, and an observer-based disturbance compensation is introduced. In contrast to the models in [2], [3], [8] and [10], where only the first bending mode is considered, the model is extended by the second bending mode.

For the experimental investigation of modern control approaches to active oscillation damping as well as tracking control, a test rig of a high-speed rack feeder has been built up at the Chair of Mechatronics at the University of Rostock, see Fig. 16.1. The experimental set-up consists of a carriage driven by an electric DC servo motor via a toothed belt, on which an elastic beam as the vertical supporting structure is mounted. On this beam structure, a cage with the load mass is guided relocatably in vertical direction. This cage with the coordinate $y_K(t)$ in horizontal direction and $x_K(t)$ in vertical direction represents the tool center point (TCP) of the rack feeder that should track desired trajectories as accurate as possible. The movable cage is driven by a tooth belt and an electric DC servo motor as well. The angles

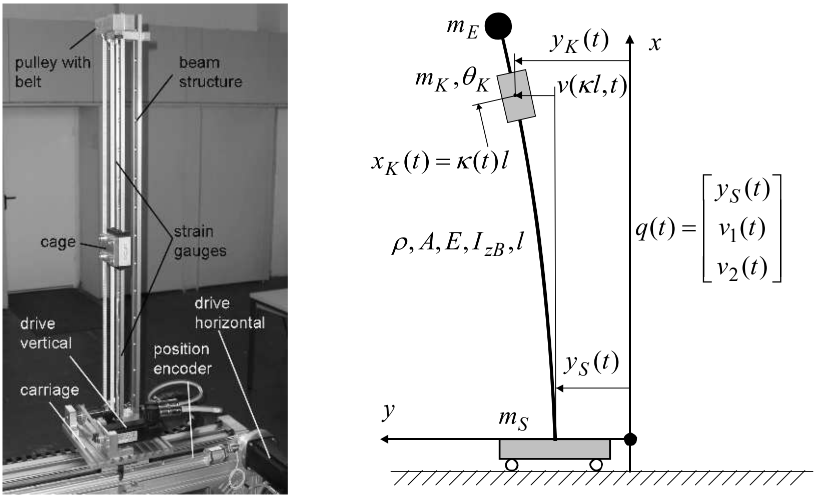


Fig. 16.1: Experimental set-up of the high-speed rack feeder (left) and the corresponding elastic multibody model (right)

of the actuators are measured by internal angular transducers, respectively. Additionally, the horizontal position of the carriage is determined by a magnetostrictive transducer. Both axes are operated with a fast underlying velocity control on the current converter. Consequently, the corresponding velocities deal as new control inputs, and the implementation effort is tremendously reduced as compared to the commonly used force or torque input, like in [10], where passivity techniques were employed for feedback control of a similar set-up. Two strain gauges are available to determine the bending deformation of the elastic beam. For the feedback of the deflection corresponding to the second bending mode and its time derivative, however, estimates from a state and disturbance observer are used.
Basis of the control design for the rack feeder, presented in Section 2, is a planar elastic multibody system, where for the mathematical description of the bending deflection of the elastic beam a Ritz ansatz is introduced, covering the first two bending modes. In Section 3, the decentralized feedforward and feedback control design for both axes is performed employing a linearized state space representation, respectively. Given couplings between both axes are taken into account by the gain-scheduling technique with the normalized vertical cage position as scheduling parameter, see also [2]. This leads to an adaptation of the whole control structure for the horizontal axis. The second bending mode, its time derivative, and a disturbance input variable are estimated by a reduced-order observer as described in Section 4. The capability of the proposed control concept is shown in Section 5 by experimental results from the test set-up with regard to tracking behavior and damping of bending oscillations. Especially the artificial damping introduced by the closed control loop represents a main improvement. The maximum velocity of the TCP during the tracking experiments is approx. 3 m/s.

16.2 Control-Oriented Modeling of the Mechatronic System

Elastic multibody models have proven advantageously for the control-oriented modeling of flexible mechanical systems. For the feedforward and feedback control design of the rack feeder a multibody model with three rigid bodies - the carriage (mass m_S), the cage movable on the beam structure (mass m_K, mass moment of inertia θ_K), and the end mass at the tip of the beam (mass m_E) - and an elastic Bernoulli beam (density ρ, cross sectional area A, Youngs modulus E, second moment of area I_{zB}, and length l) is chosen. The varying vertical position $x_K(t)$ of the cage on the beam is denoted by the dimensionless system parameter

$$\kappa(t) = \frac{x_K(t)}{l}. \tag{16.1}$$

The elastic degrees of freedom of the beam concerning the bending deflection can be described by the following Ritz ansatz

$$v(x,t) = \left[\bar{\bar{v}}_1(x)\ \bar{\bar{v}}_2(x) \right] \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix} , \tag{16.2}$$

with

$$\bar{\bar{v}}_1(x) = \frac{3}{2}\left(\frac{x}{l}\right)^2 - \frac{1}{2}\left(\frac{x}{l}\right)^3 , \tag{16.3}$$

$$\bar{\bar{v}}_2(x) = \left(\frac{x}{l}\right)^2 , \tag{16.4}$$

which takes into account the first and the second bending mode. The vector of generalized coordinates results in

$$\mathbf{q}(t) = \left[y_S(t)\ v_1(t)\ v_2(t) \right]^T . \tag{16.5}$$

The nonlinear equations of motion can be derived either by Lagrange's equations or, advantageously, by the Newton-Euler approach, cf. [9]. For this purpose, position vectors to the corresponding centers of gravity are introduced: the position vector to the carriage $\mathbf{r}_S$, to the cage $\mathbf{r}_K$, to the mass at the beam tip $\mathbf{r}_E$ and to a mass element of the Bernoulli beam $\mathbf{r}_{BE}$ are given by

$$\mathbf{r}_S = \begin{bmatrix} 0 \\ y_S \end{bmatrix} , \ \mathbf{r}_K = \begin{bmatrix} x_K \\ y_S + v(x_K) \end{bmatrix} , \ \mathbf{r}_E = \begin{bmatrix} l \\ y_S + v(l) \end{bmatrix} , \ \mathbf{r}_{BE} = \begin{bmatrix} x_{BE} \\ y_S + v(x_{BE}) \end{bmatrix} . \tag{16.6}$$

By computing the Jacobians of translation

$$\mathbf{J}_{Ti} = \frac{\partial \mathbf{r}_i}{\partial \mathbf{q}} , \ i = \{S, K, E, BE\} \tag{16.7}$$

for these vectors and the Jacobian of rotation

$$\mathbf{j}_{Rj} = \frac{\partial \varphi_j}{\partial \mathbf{q}} , \ j = \{K, E, BE\} \tag{16.8}$$

for the angles, the nonlinear equations of motion $\tilde{\mathbf{M}}(\mathbf{q})\,\ddot{\mathbf{q}} + \tilde{\mathbf{k}}(\mathbf{q}, \dot{\mathbf{q}}) = \tilde{\mathbf{h}}(\mathbf{q}, \dot{\mathbf{q}}, F_{SM}, F_{SR})$ are obtained as follows

$$\begin{aligned} \tilde{\mathbf{M}}(\mathbf{q}) =& m_S \mathbf{J}_{TS}^T \mathbf{J}_{TS} + m_K \mathbf{J}_{TK}^T \mathbf{J}_{TK} + m_E \mathbf{J}_{TE}^T \mathbf{J}_{TE} + \theta_K \mathbf{j}_{RK} \mathbf{j}_{RK}^T \\ & + \rho \int_0^l \left(A \mathbf{J}_{TBE}^T \mathbf{J}_{TBE} + I_{zB} \mathbf{j}_{RBE} \mathbf{j}_{RBE}^T \right) \mathrm{d}x , \end{aligned} \tag{16.9}$$

$$\begin{aligned} \tilde{\mathbf{k}}(\mathbf{q}, \dot{\mathbf{q}}) =& m_S \mathbf{J}_{TS}^T \dot{\mathbf{J}}_{TS} \dot{\mathbf{q}} + m_K \mathbf{J}_{TK}^T \dot{\mathbf{J}}_{TK} \dot{\mathbf{q}} + m_E \mathbf{J}_{TE}^T \dot{\mathbf{J}}_{TE} \dot{\mathbf{q}} \\ & + \theta_K \mathbf{j}_{RK} \frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{j}_{RK}^T \right) \dot{\mathbf{q}} \\ & + \rho \int_0^l \left(A \mathbf{J}_{TBE}^T \dot{\mathbf{J}}_{TBE} \dot{\mathbf{q}} + I_{zB} \mathbf{j}_{RBE} \frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{j}_{RBE}^T \right) \dot{\mathbf{q}} \right) \mathrm{d}x , \end{aligned} \tag{16.10}$$

$$\tilde{\mathbf{h}}(\mathbf{q}, \dot{\mathbf{q}}, F_{SM}, F_{SR}) = \mathbf{J}_{TS}^T \begin{bmatrix} F_{SM} - F_{SR} \\ 0 \end{bmatrix} - \frac{\partial U(\mathbf{q})}{\partial \mathbf{q}} - \frac{\partial R(\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} , \tag{16.11}$$

with the drive force of the carriage F_{SM} and the associated friction force F_{SR}. Here, the potential energy $U(\mathbf{q})$ consists of the gravity potential of all rigid and elastic bodies as well as the strain energy of the elastic beam. The Rayleigh function $R(\dot{\mathbf{q}})$ allows for an efficient computation of the stiffness-proportional damping matrix. After a linearization for small bending deflections, the equations of motion can be stated in M-D-K form

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{D}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{h} \cdot [F_{SM}(t) - F_{SR}(\dot{y}_S(t))] \ . \tag{16.12}$$

The symmetric mass matrix is given by

$$\mathbf{M}(\kappa) = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{12} & m_{22} & m_{23} \\ m_{13} & m_{23} & m_{33} \end{bmatrix} \tag{16.13}$$

with

$$m_{11} = m_S + \rho A l + m_K + m_E \ , \tag{16.14}$$

$$m_{12} = \frac{3}{8}\rho A l + \frac{m_K \kappa^2}{2}(3-\kappa) + m_E \ , \tag{16.15}$$

$$m_{13} = \frac{1}{3}\rho A l + m_K + \kappa^2 + m_E \ , \tag{16.16}$$

$$m_{22} = \frac{33}{140}\rho A l + \frac{6\rho I_{zB}}{5l} + \frac{m_K \kappa^4}{4}(3-\kappa)^2 + \frac{9\theta_K \kappa^2}{4l^2}(2-\kappa) + m_E \ , \tag{16.17}$$

$$m_{23} = \frac{13}{60}\rho A l + \frac{5\rho I_{zB}}{4l} + \frac{m_K \kappa^4}{2}(3-\kappa) + \frac{3\theta_K \kappa^2}{l^2}(2-\kappa) + m_E \ , \tag{16.18}$$

$$m_{33} = \frac{1}{5}\rho A l + \frac{4\rho I_{zB}}{3l} + m_K \kappa^4 + \frac{4\theta_K \kappa^2}{l^2} + m_E \ . \tag{16.19}$$

The damping matrix, which is assumed as stiffness-proportional, and the stiffness matrix become

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{3k_d EI_{zB}}{l^3} & \frac{3k_d EI_{zB}}{l^3} \\ 0 & \frac{3k_d EI_{zB}}{l^3} & \frac{4k_d EI_{zB}}{l^3} \end{bmatrix} , \quad \mathbf{K}(\kappa) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & k_{22} & k_{23} \\ 0 & k_{23} & k_{33} \end{bmatrix} , \tag{16.20}$$

with

$$k_{22} = \frac{3EI_{zB}}{l^3} - \frac{3}{8}\rho A g - \frac{3m_K g \kappa^3}{l}\left(1 + \frac{3\kappa^2}{20} - \frac{3\kappa}{4}\right) - \frac{6m_E g}{5l} \ , \tag{16.21}$$

$$k_{23} = \frac{3EI_{zB}}{l^3} - \frac{7}{20}\rho A g - \frac{m_K g \kappa^3}{l}\left(\frac{3\kappa}{4} - 2\right) - \frac{5m_E g}{4l} \ , \tag{16.22}$$

$$k_{33} = \frac{4EI_{zB}}{l^3} - \frac{1}{3}\rho A g - \frac{4m_K g \kappa^3}{3l} - \frac{4m_E g}{3l} \ . \tag{16.23}$$

In (16.20), the parameter k_d denotes the coefficient of stiffness-proportional damping for the elastic beam. The input vector of the generalized forces, which accounts for the control input F_{SM} as well as the disturbance input F_{SR}, reads

$$\mathbf{h} = \begin{bmatrix} 1 \; 0 \; 0 \end{bmatrix}^T . \tag{16.24}$$

The electric drive for the carriage is operated with a fast underlying velocity control on the current converter. The resulting dynamic behavior is characterized by a first-order lag system with a time constant T_{1y}

$$T_{1y}\ddot{y}_S(t) + \dot{y}_S(t) = v_S(t) - v_{S0} , \tag{16.25}$$

whereas the input disturbance v_{S0} represents remaining uncertainties. In the following, this differential equation replaces the equation of motion for the carriage in the mechanical system model (16.12), which leads to a modified mass matrix as well as a modified damping matrix

$$\mathbf{M}_y(\kappa) = \begin{bmatrix} T_{1y} & 0 & 0 \\ m_{12} & m_{22} & m_{23} \\ m_{13} & m_{23} & m_{33} \end{bmatrix} , \quad \mathbf{D}_y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{3k_dEI_{zB}}{l^3} & \frac{3k_dEI_{zB}}{l^3} \\ 0 & \frac{3k_dEI_{zB}}{l^3} & \frac{4k_dEI_{zB}}{l^3} \end{bmatrix} , \tag{16.26}$$

The stiffness matrix $\mathbf{K}_y(\kappa) = \mathbf{K}(\kappa)$ and the input vector for the generalized forces $\mathbf{h}_y = \mathbf{h}$, however, remain unchanged. Hence, the equations of motion are given by

$$\ddot{\mathbf{q}} = -\mathbf{M}_y^{-1}\mathbf{K}_y\mathbf{q} - \mathbf{M}_y^{-1}\mathbf{D}_y\dot{\mathbf{q}} + \mathbf{M}_y^{-1}\mathbf{h}_y v_S - \mathbf{M}_y^{-1}\mathbf{h}_y v_{S0} , \tag{16.27}$$

with the carriage velocity v_S as new control input u_y. For feedforward and feedback control design, a vanishing input disturbance v_{S0} is considered, and the system representation is reformulated in state space form

$$\dot{\mathbf{x}}_y = \begin{bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}_y^{-1}\mathbf{K}_y & -\mathbf{M}_y^{-1}\mathbf{D}_y \end{bmatrix}}_{\mathbf{A}_y(\kappa)} \underbrace{\begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix}}_{\mathbf{x}_y} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{M}_y^{-1}\mathbf{h}_y \end{bmatrix}}_{\mathbf{b}_y(\kappa)} \underbrace{v_S}_{u_y} . \tag{16.28}$$

The design model for the vertical movement of the cage can be directly stated in state space representation. Here, an underlying velocity control is employed on the current converter, which is also described by a first-order lag system

$$T_{1x}\ddot{x}_K(t) + \dot{x}_K(t) = v_K(t) . \tag{16.29}$$

The corresponding state space description follows immediately in the form

$$\dot{\mathbf{x}}_x = \begin{bmatrix} \dot{x}_K \\ \ddot{x}_K \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{T_{1x}} \end{bmatrix}}_{\mathbf{A}_x} \underbrace{\begin{bmatrix} x_K \\ \dot{x}_K \end{bmatrix}}_{\mathbf{x}_x} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{T_{1x}} \end{bmatrix}}_{\mathbf{b}_x} \underbrace{v_K}_{u_x} . \tag{16.30}$$

Whereas the state space representation for the horizontal y-axis depends on the varying system parameter $\kappa(t)$, the description of the x-axis is invariant. A gain-scheduling, hence, is necessary only for the horizontal axis.

16.3 Decentralized Control Design

As for control, a decentralized approach is followed, at which the coupling of the vertical cage motion with the horizontal axis is taken into account by gain-scheduling techniques. For the control of the cage position $x_K(t)$ a simple proportional feedback in combination with feedforward control, which is based on the inverse transfer function of this axis, is sufficient

$$v_K(t) = K_R \left(x_{Kd}(t) - x_K(t)\right) + \dot{x}_{Kd}(t) + T_{1x}\ddot{x}_{Kd}(t)\,. \tag{16.31}$$

For this purpose the desired trajectory $x_{Kd}(t)$ and its first two time derivatives are available from trajectory planning.

The design of the state feedback for the horizontal motion is carried out on the basis of the LQR approach, where the vector of feedback gains is determined by minimization of a quadratic cost function with a combined weighting of state variables as well as control inputs. The control gains follow from a numerical solution of the corresponding algebraic Riccati equation (ARE) in a chosen number of operating points. Due to the dependency of the system matrices on the varying system parameter κ, the ARE becomes also a function of this parameter. To guarantee the continuity of the feedback gains concerning this varying system parameter, constant weightings are employed for the states as well as the control inputs. Starting point for the LQR design is the time-weighted cost function

$$J = \frac{1}{2}\int\limits_0^\infty \left[\mathbf{x}_y^T \mathbf{Q}_y \mathbf{x}_y + r_y u_y^2\right] e^{2\alpha_y t}\, dt\,, \tag{16.32}$$

which contains the weighting matrix $\mathbf{Q}_y$, the weight r_y, and an additional parameter α_y. The weighting matrix $\mathbf{Q}_y$ for the state vector $\mathbf{x}_y$ is chosen as a constant, positive definite diagonal matrix

$$\mathbf{Q}_y = \mathrm{diag}\left[1.1e5\,, 1.5e3\,, 1.5e3\,, 40\,, 5\,, 5\right] = \mathrm{const} > 0\,, \tag{16.33}$$

the scalar input weight as constant positive value $r_y = 1e3 = \mathrm{const}$. With the parameter $\alpha_y = 2.5$ an absolute stability margin of the closed-loop eigenvalues can be specified

$$\max_i \left\{Re\left(s_i\right)\right\} \leq -\alpha_y = -2.5\,. \tag{16.34}$$

In the s-plane, hence, all closed-loop eigenvalues are located left to a parallel line in the distance α_y from the imaginary axis. In a chosen operating point, characterized by a parameter $p = \kappa$, the optimal feedback control law can be determined as positive definite solution of the parameter-dependent ARE

$$\mathbf{A}_{y\alpha}^{T}(\kappa)\,\mathbf{P}(\kappa) + \mathbf{P}(\kappa)\,\mathbf{A}_{y\alpha}(\kappa) - r_y^{-1}\mathbf{P}(\kappa)\,\mathbf{b}_y(\kappa)\,\mathbf{b}_y^{T}(\kappa)\,\mathbf{P}(\kappa) + \mathbf{Q}_y = 0\,. \tag{16.35}$$

The parameter-dependent state feedback becomes

$$u_{y,ZR}(t) = -\mathbf{k}_y^{T}(\kappa)\,\mathbf{x}_y(t)\,, \tag{16.36}$$

with the vector of feedback gains

$$\mathbf{k}_y^{T}(\kappa) = r_y^{-1}\mathbf{b}_y^{T}(\kappa)\,\mathbf{P}(\kappa)\,. \tag{16.37}$$

The matrix $\mathbf{A}_{y\alpha}$ used in the ARE is derived from the system matrix $\mathbf{A}_y$ according to

$$\mathbf{A}_{y\alpha}(\kappa) = \mathbf{A}_y(\kappa) + \alpha_y\mathbf{I}\,. \tag{16.38}$$

As the matrix $\mathbf{A}_{y\alpha}(\kappa)$ and the vector $\mathbf{b}_y(\kappa)$ continuously depend on the varying system parameter κ, and constant weightings $\mathbf{Q}_y$ and r_y are employed, the solutions $\mathbf{P}(\kappa)$ of the ARE are continuous functions of κ as well. The same applies to the vector of feedback gains. Consequently, a gain-scheduled feedback control can be derived by choosing a finite number of operating points in the space of the varying system parameter κ and by performing a LQR design with constant weightings. The resulting feedback gains contained in the vector $\mathbf{k}_y^T$ could be either approximated by simple ansatz functions, e.g. polynomials, or could be interpolated. The simplest way of implementation is given by using look-up tables with a linear interpolation between the chosen operating points, see Fig. 16.2. These look-up tables are generated automatically in advance and can be integrated in the control structure without any effort. The approximation quality can be easily specified by the number of operating points used for the control design. Obviously, a trade-off has to be found between the desired approximation quality and the necessary storage. The resulting location of the closed-loop eigenvalues in the s-plane reflecting the operating points used for the design is depicted in Fig. 16.3. The stability of the time-varying system can be shown employing a quadratic Lyapunov function

$$V(\mathbf{x}_y) = \mathbf{x}_y^T\mathbf{R}\mathbf{x}_y \tag{16.39}$$

with a constant matrix $\mathbf{R} = \mathbf{I}$. A sufficient condition for asymptotic stability is the negative definiteness of the matrix

$$\mathbf{A}_{yR} + \mathbf{A}_{yR}^T < 0\,, \tag{16.40}$$

where $\mathbf{A}_{yR} = \mathbf{A}_y - \mathbf{b}_y\mathbf{k}_y^T$ denotes the closed-loop system matrix. Alternatively, thorough simulation studies could be performed to assess the closed-loop stability. For feedforward control design the horizontal position of the cage $y_K(t)$ is considered as controlled variable. Thus, the output equation becomes

$$y_K(t) = \underbrace{\left[1\ \tfrac{1}{2}\kappa^2(3-\kappa)\ \kappa^2\ 0\ 0\ 0\right]}_{\mathbf{c}_y^T(\kappa)} \mathbf{x}_y(t)\,. \tag{16.41}$$

The control transfer function can be derived as

$$Y_K(s) = \mathbf{c}_y^T\left(s\mathbf{I} - \mathbf{A}_y + \mathbf{b}_y\mathbf{k}_y^T\right)^{-1}\mathbf{b}_y U_V(s) = \frac{\left(b_0 + b_1\cdot s + \ldots + b_4 s^4\right)}{N(s)}\cdot U_V(s)\,. \tag{16.42}$$

Obviously, the numerator of the control transfer function contains a fourth degree polynomial in s, leading to four transfer zeros. This shows that the considered output represents a non-flat output variable that makes feedforward control design more difficult. As the horizontal axis is a fully controllable linear system, which can be easily shown by computing the controllability matrix

$$\mathbf{Q}_{yS} = \left[\mathbf{b}_y\ \mathbf{A}_y\mathbf{b}_y\ \ldots\ \mathbf{A}_y^5\mathbf{b}_y\right] \tag{16.43}$$

and by checking Kalman's rank condition $\text{rank}(\mathbf{Q}_{yS}) = n = 6$, the flat system output could be computed directly

$$y_{y,fl} = a\cdot\left[0\ 0\ 0\ 0\ 0\ 1\right]^T\mathbf{Q}_{yS}^{-1}\,. \tag{16.44}$$

Unfortunately, this flat output variable does not comply with the cage position as TCP. Therefore, an alternative way is chosen to derive the feedforward control law.

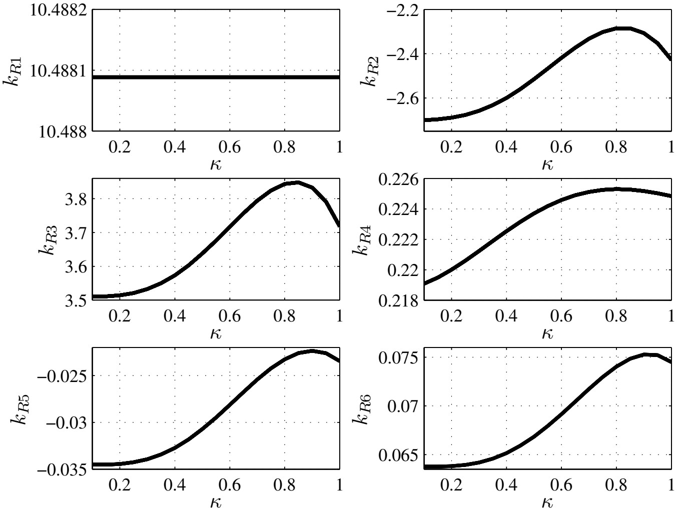


Fig. 16.2: Gain-scheduled adaption of the control gains w.r.t. the varying system parameter κ

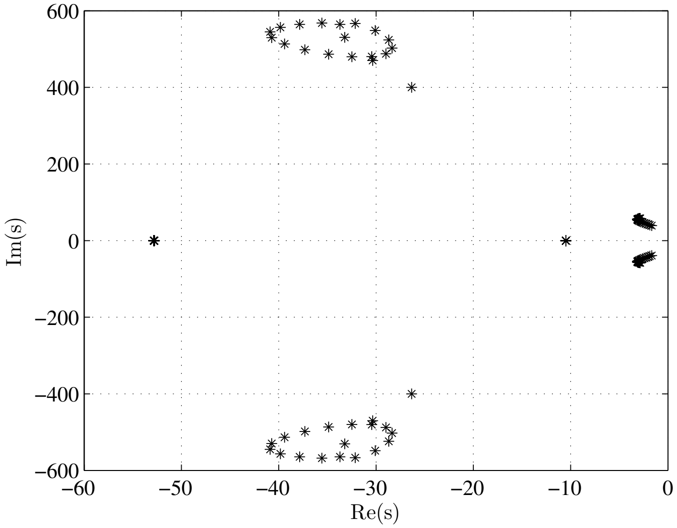


Fig. 16.3: Closed-loop eigenvalues in dependency on the varying system parameter κ

The main idea is given by a modification of the numerator of the control transfer function by introducing a polynomial ansatz for the feedforward control action according to

$$U_V(s) = \left[k_{V0} + k_{V1} \cdot s + \ldots + k_{V4} \cdot s^4\right] Y_{Kd}(s) \; . \tag{16.45}$$

For its realization the desired trajectory $y_{Kd}(t)$ as well as the first four time derivatives are available from a trajectory planning module. The feedforward gains can be computed from a comparison of the corresponding coefficients in the numerator as well as the denominator polynomials of

$$\begin{aligned}\frac{Y_K(s)}{Y_{Kd}(s)} &= \frac{\left(b_0 + \ldots + b_4 \cdot s^4\right)\left[k_{V0} + \ldots + k_{V4} \cdot s^4\right]}{N(s)} \\ &= \frac{b_{V0}(k_{Vj}) + b_{V1}(k_{Vj}) \cdot s + \ldots + b_{V8}(k_{Vj}) \cdot s^8}{a_0 + a_1 \cdot s + \ldots + s^6}\end{aligned} \tag{16.46}$$

according to

$$a_i = b_{Vi}(k_{Vj}) \, , i = 0, \ldots, 4 \, . \tag{16.47}$$

This leads to parameter-dependent feedforward gains $k_{Vj} = k_{Vj}(\kappa)$. It becomes obvious that due to the higher numerator degree in the modified control transfer function a remaining dynamics must be accepted. Though perfect tracking could not be achieved due to the transfer zeros of the open-loop system, this easily implementable feedforward control contributes significantly to an improved tracking behavior.

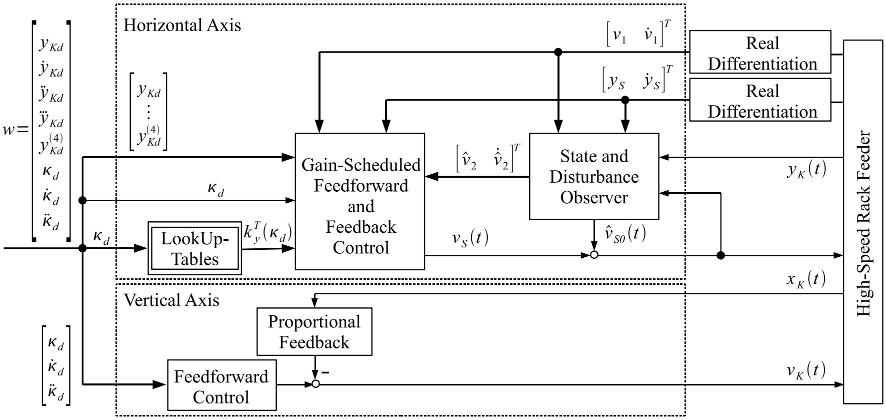

Horizontal Axis
$w=$
$\begin{bmatrix} y_{Kd} \\ \dot{y}_{Kd} \\ \ddot{y}_{Kd} \\ \dddot{y}_{Kd} \\ y_{Kd}^{(4)} \\ \kappa_d \\ \dot{\kappa}_d \\ \ddot{\kappa}_d \end{bmatrix}$
$\begin{bmatrix} y_{Kd} \\ \vdots \\ y_{Kd}^{(4)} \end{bmatrix}$
κ_d
κ_d
LookUp-Tables
$k_y^T(\kappa_d)$
Gain-Scheduled Feedforward and Feedback Control
$\begin{bmatrix} \hat{v}_2 & \hat{\dot{v}}_2 \end{bmatrix}^T$
State and Disturbance Observer
$\begin{bmatrix} v_1 & \dot{v}_1 \end{bmatrix}^T$
$\begin{bmatrix} y_S & \dot{y}_S \end{bmatrix}^T$
Real Differentiation
Real Differentiation
$v_S(t)$
$\hat{v}_{S0}(t)$
$y_K(t)$
$x_K(t)$
$v_K(t)$
High-Speed Rack Feeder
Vertical Axis
$\begin{bmatrix} \kappa_d \\ \dot{\kappa}_d \\ \ddot{\kappa}_d \end{bmatrix}$
Proportional Feedback
Feedforward Control
-

Fig. 16.4: Implementation of the control structure

16.4 State and Disturbance Observer Design

To obtain estimates of the bending deflection v_2 and its time derivative $\dot{v}_2$ as well as the disturbance variable v_{S0}, a state and disturbance observer is employed as described in [5]. The observer design is based on the equations of motion. The key idea for the observer design is to divide the state-space representation in measurable and non-measurable subsystems and extend them with an integrator as disturbance model

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{f}_1\left(\mathbf{x}_1, \mathbf{x}_2, v_{S0}, \mathbf{u}\right) \ , \\ \dot{\mathbf{x}}_2 &= \mathbf{f}_2\left(\mathbf{x}_1, \mathbf{x}_2, v_{S0}, \mathbf{u}\right) \ , \\ \dot{v}_{S0} &= 0 \ , \end{aligned} \tag{16.48}$$

where $\mathbf{x}_1 = \begin{bmatrix} y_S \ v_1 \ \dot{y}_S \ \dot{v}_1 \end{bmatrix}^T$ comprises the measurable states and $\mathbf{x}_2 = \begin{bmatrix} v_2 \ \dot{v}_2 \end{bmatrix}^T$ contains the non-measurable state variables. The vector including the estimated states and the estimated disturbance variable $\hat{\mathbf{x}}_O = \begin{bmatrix} \hat{v}_2 \ \hat{\dot{v}}_2 \ \hat{v}_{S0} \end{bmatrix}^T$ is obtained from

$$\hat{\mathbf{x}}_O = \mathbf{H}\mathbf{x}_1 + \mathbf{z} \tag{16.49}$$

with the chosen observer gain matrix

$$\mathbf{H} = \begin{bmatrix} 0 \ 0 \ 0 \ h_{14} \\ 0 \ 0 \ 0 \ h_{24} \\ 0 \ 0 \ 0 \ h_{34} \end{bmatrix} \ . \tag{16.50}$$

The state equation for $\mathbf{z}$ is given by

$$\dot{\mathbf{z}} = \boldsymbol{\Phi}\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) \ . \tag{16.51}$$

The observer gain matrix $\mathbf{H}$ and the vector function $\boldsymbol{\Phi}$ have to be chosen such that the steady-state observer error $\mathbf{e} = \mathbf{x}_O - \hat{\mathbf{x}}_O$, $\mathbf{x}_O = \left[v_2 \ \dot{v}_2 \ v_{S0} \right]^T$ converges to zero. Thus, the vector function $\boldsymbol{\Phi}$ can be determined as follows

$$\dot{\mathbf{e}} = \mathbf{0} = \begin{bmatrix} \mathbf{f}_2\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) \\ 0 \end{bmatrix} - \mathbf{H}\mathbf{f}_1\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) - \boldsymbol{\Phi}\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) \ . \qquad (16.52)$$

This yields an equation for calculation of the observer term $\boldsymbol{\Phi}\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right)$

$$\boldsymbol{\Phi}\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) = \begin{bmatrix} \mathbf{f}_2\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) \\ 0 \end{bmatrix} - \mathbf{H}\mathbf{f}_1\left(\mathbf{x}_1, \hat{\mathbf{x}}_O, u\right) \ . \qquad (16.53)$$

The error dynamics $\dot{\mathbf{e}}$ has to be asymptotically stable. Accordingly, all eigenvalues of the Jacobian

$$\mathbf{J}_e = \frac{\partial \boldsymbol{\Phi}\left(\mathbf{x}_1, \mathbf{x}_O, u\right)}{\partial \mathbf{x}_O} \qquad (16.54)$$

must be located in the left complex half-plane. This can be achieved by proper choice of the observer gains h_{14}, h_{24} and h_{34}. The stability of the closed-loop control system has been investigated by thorough simulations. The control implementation including the reduced-order observer is illustrated in Fig. 16.4.

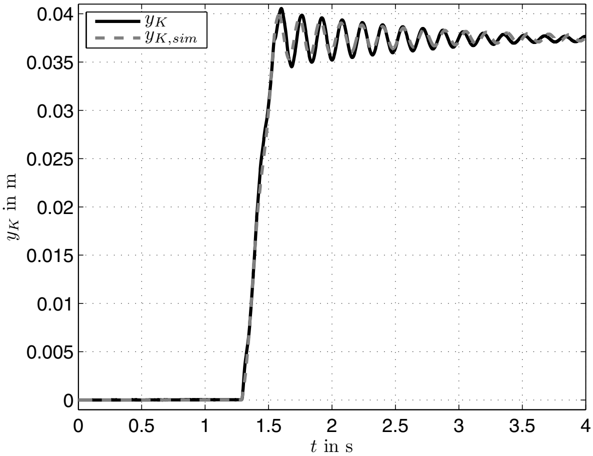


Fig. 16.5: Comparison of the response of y_K and $y_{K,sim}$ to a rectangle pulse using the identified parameters

16.5 Parameter Identification

A large number of the system parameters are given by material properties or can be determined theoretically from geometrical considerations. These parameters are listed in Tab. 16.1. The remaining parameters T_{1y}, I_{zB}, k_d as well as T_{1x} are determined experimentally using parameter identification techniques. For this purpose, the cage positions x_K and y_K are measured at the test rig for a given input signals v_S and v_K. Then, the simulation model is evaluated with the same input signals, and the obtained output signals $x_{K,sim}$ and $y_{K,sim}$ are compared to the corresponding measured outputs x_K and y_K. The identification task to be solved, hence, consists in the minimization of the quadratic cost functions

Table 16.1: Known system parameters

Length of the beam	l	1.07 m
Cross sectional area of the beam	A	$6e^{-4}$ m^2
Youngs modulus of the beam	E	$70e^9$ $\frac{\mathrm{N}}{\mathrm{m}^2}$
Density of the beam	ρ	$2.7e^3$ $\frac{\mathrm{kg}}{\mathrm{m}^3}$
Mass at the tip of the beam	m_E	0.9 kg
Mass of the cage	m_K	0.95 kg

$$J\left(T_{1y}, I_{zB}, k_d\right) = \sum_{i=1}^{N} \left[y_{K,i} - y_{K,sim,i}\left(T_{1y}, I_{zB}, k_d\right)\right]^2 \tag{16.55}$$

and

$$J\left(T_{1x}\right) = \sum_{i=1}^{N} \left[x_{K,i} - x_{K,sim,i}\left(T_{1x}\right)\right]^2 \ , \tag{16.56}$$

where N denotes the number of signal samples. For the nonlinear optimization the Matlab function `fminsearch` has been used, which is based on the Nelder-Mead approach. A comparison of the measured cage position y_K and the simulated position $y_{K,sim}$ using the identified parameters is shown in Fig. 16.5 for a rectangle pulse as input variable $u_y = v_S$. The identified parameters are stated in Tab. 16.2.

Table 16.2: Identified system parameters

Time constant x-direction	T_{1x}	0.013 s
Time constant y-direction	T_{1y}	0.007 s
Second moment of area of the beam	I_{zB}	$1.2e^{-8}$ m^4
Damping coefficient of the beam	k_d	$1.1e^{-3}$ s

16.6 Experimental Validation on the Test Rig

The benefits and the efficiency of the proposed control measures shall be pointed out by experimental results obtained from the test set-up available at the Chair of Mechatronics, University of Rostock.

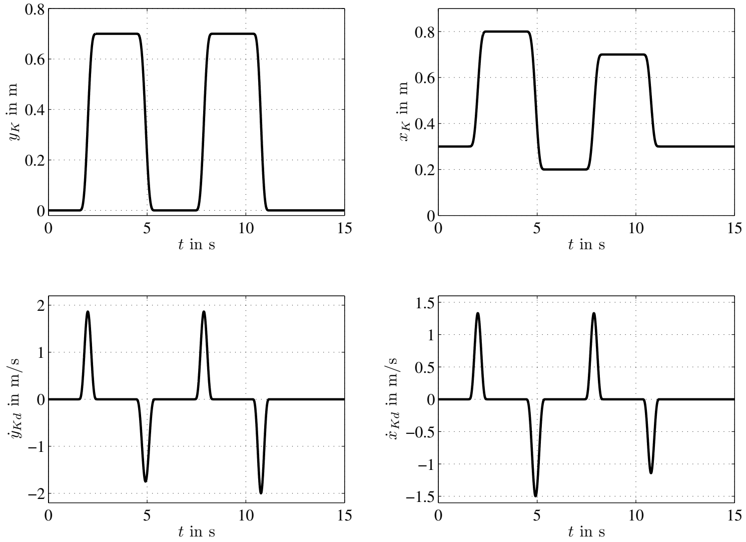


Fig. 16.6: Desired trajectories for the cage motion: desired position in horizontal direction (upper left part), desired position in vertical direction (upper right part), desired velocity in horizontal direction (lower left part) and desired velocity in vertical direction (lower right part)

For this purpose, a synchronous, four times continuously differentiable desired trajectory is considered for the position of the cage in both x- and y-direction. The desired trajectory is given by polynomial functions that comply with specified kinematic constraints, which is achieved by taking advantage of time-scaling techniques. The desired trajectory shown in Fig. 16.6 comprises a sequence of reciprocating motions with maximum velocities of 2 m/s in horizontal direction and 1.5 m/s in vertical direction. The resulting tracking errors

$$e_y(t) = y_{Kd}(t) - y_K(t) \tag{16.57}$$

and

$$e_x(t) = x_{Kd}(t) - x_K(t) \tag{16.58}$$

are depicted in Fig. 16.7. As can be seen, the maximum position error during the movements is about 2 mm in y-direction and about 3 mm in x-direction. The steady-state position error in y- as well as in x-direction is smaller than 0.3 mm. Fig. 16.8 shows the comparison of the bending deflection measured by strain gauges attached to the flexible beam with desired values as well as the difference between these two signals $e_{v1} = v_{1d} - v_1$. The desired values of the system states can be derived from the available inverse system model, for example in the Laplace domain. For this purpose, the polynomial feedforward law $U_V(s)$ has to be partially differentiated with respect to the corresponding feedback gain. The desired bending deflections v_{1d} and v_{2d} are obtained as follows

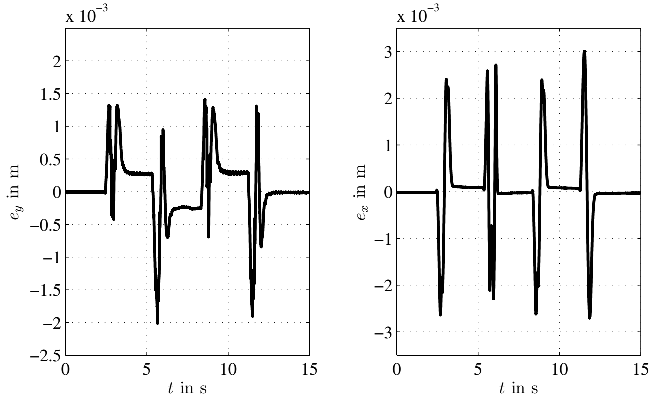


Fig. 16.7: Tracking error $e_y(t)$ for the cage motion in horizontal direction (left part) and tracking error $e_x(t)$ for the cage motion in vertical direction (right part)

$$v_{1d}(s) = \frac{\partial U_V(s)}{\partial k_{R2}} = \left[\frac{\partial k_{V0}}{\partial k_{R2}} + \frac{\partial k_{V1}}{\partial k_{R2}} \cdot s + \ldots + \frac{\partial k_{V4}}{\partial k_{R2}} \cdot s^4\right] Y_{Kd}(s) \ , \tag{16.59}$$

$$v_{2d}(s) = \frac{\partial U_V(s)}{\partial k_{R3}} = \left[\frac{\partial k_{V0}}{\partial k_{R3}} + \frac{\partial k_{V1}}{\partial k_{R3}} \cdot s + \ldots + \frac{\partial k_{V4}}{\partial k_{R3}} \cdot s^4\right] Y_{Kd}(s) \ . \tag{16.60}$$

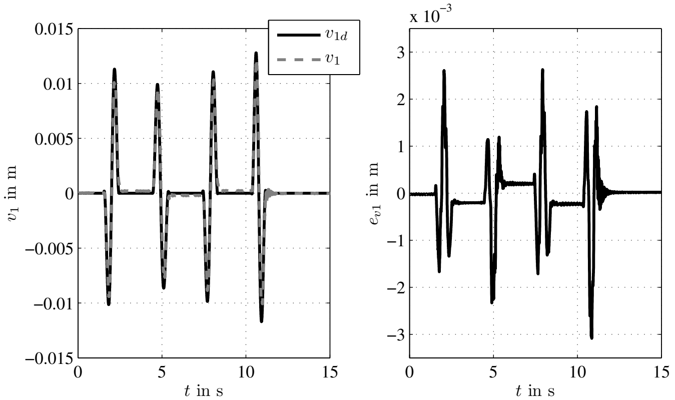


Fig. 16.8: Comparison of the desired values $v_{1d}(t)$ and the actual values $v_1(t)$ for the bending deflection (left part) and difference $e_{v1}(t) = v_{1d}(t) - v_1(t)$ between the values $v_{1d}(t)$ and $v_1(t)$ (right part)

During the acceleration as well as the deceleration intervals, physically unavoidable bending deflections could be noticed. The achieved benefit is given by the fact that the remaining oscillations are negligible when the rack feeder arrives at its target position. This underlines both the high model accuracy and the quality of the active

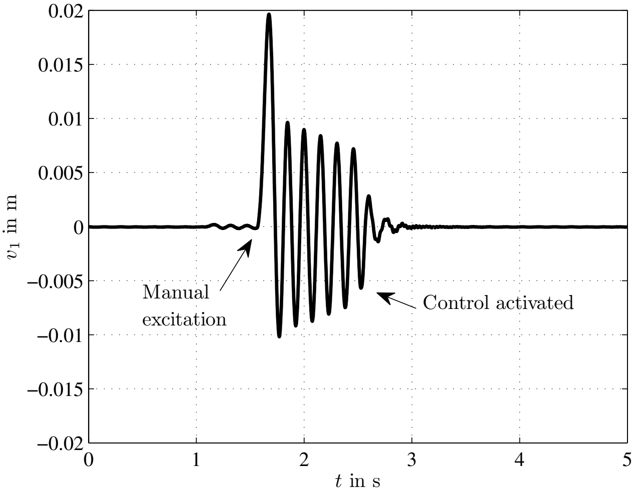


Fig. 16.9: Transient response after a manual excitation of the bending deflection: at first without feedback control, after approx. 2.5 seconds with active control

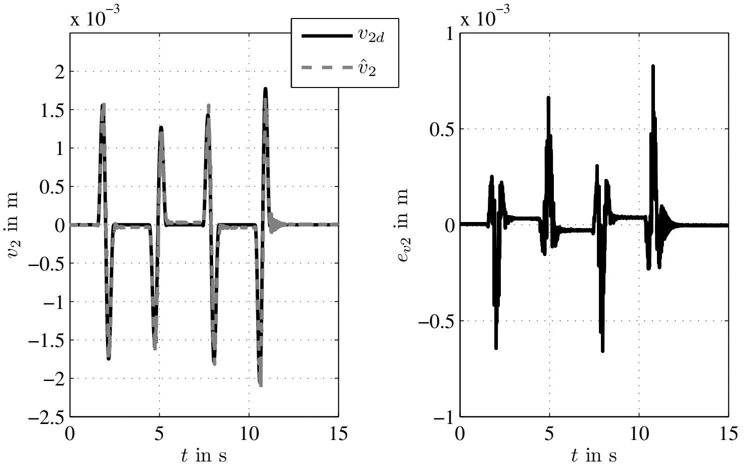


Fig. 16.10: Comparison of the desired value $v_{2d}(t)$ and the estimated value $\hat{v}_2(t)$ (left part); difference $e_{v2}(t) = v_{2d}(t) - \hat{v}_2(t)$ between the desired value $v_{2d}(t)$ and the estimated value $\hat{v}_2(t)$ (right part)

damping of the first two bending modes. Fig. 16.9 depicts the disturbance rejection properties due to an external excitation by hand. At the beginning, the control structure is deactivated, and the excited bending oscillations decay only due to the very low material damping. After approx. 2.5 seconds, the control structure is activated and, hence, the bending oscillations are actively damped. The estimated values for the second bending mode $\hat{v}_2$ are shown in the left part of Fig. 16.10. Here $\hat{v}_2$ is compared to the desired bending deflection v_{2d}. The right part of Fig. 16.10 shows the error $e_{v2} = v_{2d} - \hat{v}_2$ between the desired and the estimated bending defection. As can be seen the estimated values are in good consistency with the desired values. In the left part of Fig. 16.11 the estimated disturbance velocity $\hat{v}_{S0}$ is depicted. The impact of the observer-based disturbance compensation strategy can be seen in the right part of Fig. 16.11. Without disturbance compensation the maximum tracking error increases up to approx. 8 mm.

16.7 Conclusions

In this paper, a gain-scheduled feedforward and feedback control strategy for flexible high-speed light-weight rack feeders is presented. The control design is based on a control-oriented elastic multibody system. The state variables concerning the second bending mode as well as an input disturbance are estimated by a reduced-order observer. Beneath an active oscillation damping of the first two bending modes, an accurate trajectory tracking for the cage position in x- and y-direction is achieved.

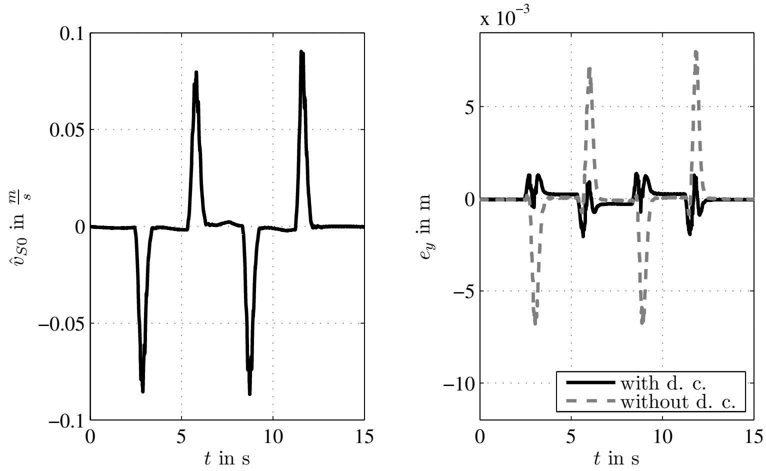


Fig. 16.11: Estimated disturbance velocity $\hat{v}_{S0}(t)$ (left part); comparison of the tracking errors $e_y(t)$ with and without disturbance compensation (d. c.) (right part)

Experimental results from a prototypic test set-up point out the benefits of the proposed control structure. Experimental results show maximum tracking errors of approx. 3 mm in transient phases, whereas the steady-state tracking error is smaller than 0.3 mm. Future work will address an adaption of the control structure to changing masses of the payload.

References

1. Aschemann, H., Ritzke, J.: Adaptive aktive Schwingungsdämpfung und Trajektorienfolgeregelung für hochdynamische Regalbediengeräte. In: Schwingungen in Antrieben, Vorträge der 6. VDI-Fachtagung in Leonberg, Germany (2009). (in German)
2. Aschemann, H., Ritzke, J.: Gain-scheduled tracking control for high-speed rack feeders. Proc. of the first joint international conference on multibody system dynamics (IMSD), 2010, Lappeenranta, Finland (2010)
3. Aschemann, H., Schindele, D.: Model predictive trajectory control for high-speed rack feeders. In: T. Zheng (ed.) Model Predictive Control. Sciyo (2010)
4. Bachmayer, M., Rudolph, J., Ulbrich, H.: Flatness based feed forward control for a horizontally moving beam with a point mass. European Conference on Structural Control, St. Petersburg, Russia pp. 74–81 (2008)
5. Friedland, B.: Advanced Control System Design. Prentice-Hall (1996)
6. Kostin, G.V., Saurin, V.V.: The Optimization of the Motion of an Elastic Rod by the Method of Integro-Differential Relations. In: Journal of computer and Systems Sciences International, vol. 45, pp. 217–225. Pleiades Publishing, Inc. (2006)
7. M. Bachmayer, J.R., Ulbrich, H.: Acceleration of linearly actuated elastic robots avoiding residual vibrations. In: Proceedings of the 9th International Conference on Motion and Vibration Control, Munich, Germany (2008)

8. Schindele, D., Aschemann, H., Ritzke, J.: Norm-optimal ILC applied to a high-speed rack feeder. 8th International Conference of Numerical Analysis and Applied Mathematics (ICNAAM), 2010, Rhodes, Greece (2010)
9. Shabana, A.A.: Dynamics of multibody systems. Cambridge University Press, Cambridge (2005)
10. Staudecker, M., Schlacher, K., Hansl, R.: Passivity based control and time optimal trajectory planning of a single mast stacker crane. Proc. of the 17th IFAC World Congress, Seoul, Korea pp. 875–880 (2008)

Notation

As a basis for the contributions to this Special Volume on *Modeling, Design, and Simulation of Systems with Uncertainties*, we have agreed with all authors on the following set of notations which is used as far as possible. As a general convention, small letters are reserved for scalars (e.g. a) and vectors (e.g. $\mathbf{x}$) and capital letters (e.g. $\mathbf{A}$) for matrices. Vectors and matrices are typeset in boldface letters.

Special Sets and Operators

$\mathbb{R}$	Set of real numbers
$\mathbb{R}^n$	n-dimensional real Euclidean space
$\mathbb{IR}$	Set of all scalar real intervals, i.e., $\mathbb{IR} := \{\, [x] \vert \underline{x} \leq \overline{x} \text{ for all } \underline{x}, \overline{x} \in \mathbb{R}\}$
$\mathbb{S}$	An arbitrary set of real/ complex values
$\emptyset$	Empty set

Specification of Interval Variables

$[x]$	Interval enclosure of a scalar real variable $x \in \mathbb{R}$
$[\mathbf{x}]$	Interval enclosure of a real vector $\mathbf{x} \in \mathbb{R}^n$
$\underline{x}, \inf\{[x]\}$	Infimum (lower bound) of an interval $[x] \in \mathbb{IR}$
$\overline{x}, \sup\{[x]\}$	Supremum (upper bound) of an interval $[x] \in \mathbb{IR}$
$[x] \circ [y]$	Interval operations for scalar operands $[x]$, $[y]$, $\circ \in \{+, -, \cdot, /\}$

$$\begin{aligned}[z]{:=}[x] \circ [y] &:= \{x \circ y \mid x \in [x]\,, y \in [y]\}\\ &= \{z \mid \min(\underline{x} \circ \underline{y}, \underline{x} \circ \overline{y}, \overline{x} \circ \underline{y}, \overline{x} \circ \overline{y}) \leq z \leq \max(\underline{x} \circ \underline{y}, \underline{x} \circ \overline{y}, \overline{x} \circ \underline{y}, \overline{x} \circ \overline{y})\}\end{aligned}$$

For the division of intervals, the case $0 \in [y]$ needs special treatment

A. Rauh and E. Auer, *Modeling, Design, and Simulation of Systems with Uncertainties,*
Mathematical Engineering 3, DOI 10.1007/978-3-642-15956-5,

$\operatorname{diam}\{[x]\}$	Diameter of an interval $[x]$, $\operatorname{diam}\{[x]\} := \overline{x} - \underline{x}$
$\operatorname{mid}\{[x]\}$	Midpoint of an interval $[x]$, $\operatorname{mid}\{[x]\} := \frac{1}{2}\left(\underline{x} + \overline{x}\right)$
$\operatorname{rad}\{[x]\}$	Radius of an interval $[x]$, $\operatorname{rad}\{[x]\} := \frac{1}{2}\left(\overline{x} - \underline{x}\right)$
$\operatorname{vol}\{[\mathbf{x}]\}$	Pseudo volume of an interval box $[\mathbf{x}]$ with $\operatorname{vol}\{[\mathbf{x}]\} := \prod_{i=1}^{n} \operatorname{diam}\{[x_i]\}$, $\mathbf{x} \in \mathbb{R}^n$
$f([x])$	Any interval evaluation of the function f over the interval $[x]$
$V_f([x])$	Range of the function $f : \mathbb{R} \mapsto \mathbb{R}$ over the interval $[x] \in \mathbb{IR}$, i.e., $V_f([x]) := \{f(x) \mid x \in [x]\}$
$\subseteq$	Subset, $[a] \subseteq [b]$ means $x \in [b]$ for all $x \in [a]$
$\subset$	*True* subset, i.e., $[a] \subset [b]$ means $[a] \subseteq [b]$ and $[a] \neq [b]$
$\in$	Element of a set
$\cup$	Union operator for sets
$\sqcup$	Convex hull operation for intervals, i.e., $[\mathbf{x}] \sqcup [\mathbf{y}] := \left[\min\left\{\underline{\mathbf{x}}, \underline{\mathbf{y}}\right\} ; \max\left\{\overline{\mathbf{x}}, \overline{\mathbf{y}}\right\}\right]$
$\cap$	Intersection operator for sets and intervals, i.e.,

$$[\mathbf{x}] \cap [\mathbf{y}] := \begin{cases} \left[\max\left\{\underline{\mathbf{x}}, \underline{\mathbf{y}}\right\} ; \min\left\{\overline{\mathbf{x}}, \overline{\mathbf{y}}\right\}\right] & \text{for } \max\left\{\underline{x}_i, \underline{y}_i\right\} \leq \min\left\{\overline{x}_i, \overline{y}_i\right\}, \; i = 1, \ldots, n \\ \emptyset & \text{else} \end{cases}$$

For multi-dimensional interval boxes, $\inf\{[\mathbf{x}]\}$, $\sup\{[\mathbf{x}]\}$, $\operatorname{diam}\{[\mathbf{x}]\}$, $\operatorname{mid}\{[\mathbf{x}]\}$, $\operatorname{rad}\{[\mathbf{x}]\}$ are defined element-wise, e.g., $\overline{\mathbf{x}} = \sup\{[\mathbf{x}]\}$ with $\overline{x}_i = \sup\{[x_i]\}$

Specification of Stochastic Variables

ω	Event, outcome of a random experiment
$(\Omega, \mathscr{F}, \mathbf{P})$	Probability space
Ω	Sample space, set of all possible outcomes
$\mathscr{F}$	Set of events
$\mathbf{P}$	Probability measure
$\mathrm{E}[y]$	Expectation of y
$\operatorname{Cov}(x,y)$	Covariance $\operatorname{Cov}(x,y) \triangleq \mathrm{E}\left[(x - \mathrm{E}[y])(x - \mathrm{E}[y])\right]$
$\operatorname{Var}[y]$	Variance of y
$\mathrm{E}[y \mid \mathscr{G}]$	Conditional expectation of y given $\mathscr{G}$
$p(t,y)$	Probability density
$Y(t)$	Stochastic process